

DIPLOMARBEIT

**DIGITALE REGELUNG EINES
PERMANENTMAGNET-SYNCHRONANTRIEBES**

eingereicht bei:

Prof. Dr.-Ing. D. Naunin
- Institut für Elektronik -
Technische Universität Berlin

Dieter Hetzel
Strelitzstr. 1
1000 Berlin 42

INHALTSVERZEICHNIS

	Seite
Einleitung	1
1. DIE THEORIE DER PM-SYNCHRONMASCHINE	
1.1 Voraussetzungen und Bezeichnungen	3
1.2 Das Maschinenmodell	4
1.3 Raumzeigerdarstellung	7
1.4 Spannungsgleichung in Raumzeigerdarstellung	8
1.5 Rotorfestes Koordinatensystem	9
1.6 Systemgleichungen im rotorfesten System	10
1.7 Systemgleichungen im statorfesten System	11
1.8 Bezogenes Größensystem	11
1.9 Umrechnung der Maschinengleichungen in bezogene Größen	15
1.10 Linearisierung des Maschinenmodells	17
2. ZEITDISKRETES MASCHINENMODELL	
2.1 Herleitungsverfahren	20
2.2 Zeitdiskretes Maschinenmodell mit Berücksichtigung der Reibung	21
2.3 Zeitdiskretes Maschinenmodell bei Vernachlässigung der Reibung	25
3. Z-TRANSFORMATION UND REGLERENTWURFSMETHODEN	
3.1 Die Z-Transformation	28
3.2 Eigenschaften der Z-Transformation	28
3.3 Inverse Z-Transformation	31
3.4 Eigenschaften der Z-ÜF digitaler Regler	32
3.5 Z-ÜF einfacher digitaler Regler	35
3.6 Z-Übertragungsfunktion und Systemeigenschaften	36
3.7 Reglerentwurf mit der Z-Übertragungsfunktion	
3.7.1 Die Z-Übertragungsfunktion der Strecke	40
3.7.2 Die Führungs-Übertragungsfunktion	41
3.7.3 Entwurf eines Einfachregelkreises	42
3.7.4 Entwurf eines P-Zustandsreglers	43

3.7.5 Entwurf eines PI-Zustandsreglers	46
3.7.6 Entwurf eines Zustandsreglers mit Totzeitberücksichtigung	52
3.7.7 Zustandsreglerentwurf bei einer Strecke mit Blockstruktur	53
3.7.8 Entwurf eines kaskadierten Reglers	56
 4. STROMREGLERSTRUKTUREN	
4.1 Allgemeines	57
4.2 Stromsteuerung	58
4.3 Zeitkontinuierliche Stromregelung	
4.3.1 P-Stromregelung ohne Störgrößenkompensation	61
4.3.2 P-Stromregelung mit Kompensation der Störgrößen	62
4.3.3 P-Stromzustandsregler	64
4.4 Zeitdiskrete Stromregelung ohne Totzeit	
4.4.1 Allgemeines	64
4.4.2 Stromzustandsregler im rotorfesten System	65
4.4.3 Stromzustandsregler im statorfesten System	66
4.4.4 Vergleich der Reglerstrukturen	67
4.4.5 Stromzustandsregler im gedrehten statorfesten System	69
4.5 Stromreglerstrukturen mit Totzeitberücksichtigung	
4.5.1 Allgemeines	70
4.5.2 Stromzustandsregler im rotorfesten System	71
4.5.3 Stromzustandsregler mit Zwischenzustandsberechnung	73
4.5.5 Stromzustandsregler im statorfesten System	73
4.5.6 Vergleich der Stromzustandsregler	75
 5. REGLERSTRUKTUREN	
5.1 Allgemeines	76
5.2 Reglerstrukturen ohne Berücksichtigung der Totzeit	
5.2.1 Klassische Kaskadenreglerstruktur	77
5.2.2 Zustandsreglerstruktur	79
5.2.3 Vergleich der Reglerstrukturen	82
5.2.4 Kaskadierter Zustandsregler	87
5.3 Reglerstrukturen mit Berücksichtigung der Totzeit	
5.3.1 Allgemeines	89
5.3.2 Beobachterfunktionen	89

5.3.3 Zustandsregler mit Totzeitberücksichtigung	90
5.3.4 Kaskadierter Zustandsregler mit Totzeit- berücksichtigung	91
5.3.5 Kaskadenregler mit Totzeitberücksichtigung	91
6. ENTWURF DER REGLER	
6.1 Übertragungsfunktionen der Regelstrecke	93
6.2 Entwurf des Stromreglers	96
6.3 Entwurf eines Drehzahlzustandsreglers	98
6.3 Entwurf des Lagezustandsreglers	100
7. SIMULATION DER REGELUNG	
7.1 Zielstellung	102
7.2 Programmiersystem	102
7.3 Grobstruktur des Programms - Menues	103
7.4 Beschreibung der Menues und Optionen	105
7.5 Implementierte Stromregler	110
7.6 Implementierte übergeordnete Regler	112
7.7 Datenstruktur des Programms	114
7.8 Der Modul Simulation	114
8. IMPLEMENTIERUNG DER REGELUNG	
8.1 Der Prozessor 8096	117
8.2 Das Entwicklungssystem SBE96	118
8.3 Die Struktur des Assemblerprogramms	
8.3.1 Bezeichnungen	119
8.3.2 Programmebenen	119
8.3.3 Die Datenstruktur	121
8.3.4 Datenformat der Variablen	122
8.3.5 Rechnungen im Festkommaformat	124
8.3.6 Datenformat der Reglerkonstanten	126
8.4 Die Interruptroutine MOT_INT	
8.4.1 Struktur der Interruptroutine	127
8.4.2 Die Routine SPSYNTH	128
8.4.3 Die Routine CAMFILL	129
8.5 Die Routine REGELUNG	
8.5.1 Die Struktur der Routine	130
8.5.2 Testroutinen	134
8.5.2 Lage- und Drehzahlregler	136

8.5.3	Stromreglerrouninen	138
8.6	Die Kommunikationsebene	
8.6.1	Struktur der Kommunikationsebene	139
8.6.2	Befehlseingabe	140
8.3.3	Die Routine MESS_OUT	140
8.3.4	Der Befehlsinterpreter COM_INTERPR	141
8.3.5	Die Sollwertverwaltung	141
8.3.6	Befehlsübersicht	142
8.7	Beschreibung einzelner Rechenroutinen	
8.7.1	Die Routine REPOL	145
8.7.2	Die Routine XYDQ	148
8.7.3	Die Routine POLRE	149
8.7.4	Lageauswertung	149
8.7.5	Drehzahlberechnung	151
8.7.6	Die Begrenzeroutine LIMIT	152
8.7.7	Kompensationsberechnungen	153
8.8	Rechenzeiten	153
8.9	Die Hilfsprogramme PARAGEN und OBJCONV	
8.9.1	Das Programm PARAGEN	153
8.9.2	Das Programm OBJCONV	154
9.	SIMULATIONS- UND MESSERGEBNISSE	
9.1	Maschinenparameter	156
9.2	Systemverhalten	157
9.3	Stromregelung und Stromsteuerung	158
9.4	Lageregelung	159
Anhang:	Programmdokumentation	171
Literaturverzeichnis		173

EINLEITUNG

Diese Arbeit behandelt Methoden zur vollständig digitalen Lage- und Drehzahlregelung permanentmagnetisch erregter Synchronmotoren. Diese werden im folgenden auch als PM-Synchronmotoren bezeichnet.

Herkömmliche Regelungen besitzen eine hybride Struktur mit einer digitalen Lageregelung und einer analogen Drehzahl- und Stromregelung. Die Fortschritte auf dem Gebiet der Mikroprozessortechnik ermöglichen nun vollständig digitale Regelstrukturen. Vollständig digitale Regelalgorithmen zur Lageregelung benötigen bei Verwendung eines Einprozessorsystems beim gegenwärtigen Stand der Technik Rechenzeiten von ca. 1 ms.

Es werden Synchronmotoren der Fa. MAVILOR verwendet, die mit Samarium-Kobalt-Permanentmagneten ausgestattet sind. Diese PM-Synchronmotoren besitzen bei einer Leistung im Bereich von 1 kW elektrische Zeitkonstanten T_{el} von ca. 2 bis 3 ms. Die mechanische Zeitkonstante T_{mech} der Maschinen ist dabei nur wenig größer als die elektrische Zeitkonstante.

Das System arbeitet mit einem Gleichspannungszwischenkreis. Die Zwischenkreisspannung beträgt ca. 80 V. Als Schaltelemente werden MOS-FETs verwendet. Die Spannungszeigersynthese erfolgt nach dem in /4 HETZEL/ beschriebenen 3-Vektor-Verfahren mit der Taktzeit $T_{PWR} = 128 \mu s$. Bei der gewählten Taktzeit kann die Spannungszeigersynthese als kontinuierlich wirkend betrachtet werden. Die Eigenschaften des Syntheseverfahrens werden deshalb bei der Regelung nicht berücksichtigt.

Neben der Untersuchung der Eigenschaften verschiedener Reglerstrukturen wurde im Rahmen dieser Arbeit ein PC-Programm zur Simulation der Regelung erstellt, sowie die Implementation der Regelung auf dem gegebenen Mikroprozessorsystem MC 8096 durchgeführt.

Das mathematische Modell und das stationäre und dynamische Verhalten der PM-Synchronmaschine sind vom Autor dieser Arbeit ausführlich in /4/ dargestellt worden. Diese Arbeit baut auf den dort erzielten Resultaten auf.

In Kapitel 1 dieser Arbeit wird das mathematische Modell der PM-Synchronmaschine dargestellt. Die Darstellung erfordert die Einführung von Raumzeigern und die Verwendung des rotorfesten Bezugssystems. Wie in /4/ werden bezogene Größen verwendet. Die Einführung dieser bezogenen Größen erwies sich als vorteilhaft, da sie zu einfacheren Gleichungen führt. Die Herleitung der bezogenen Größen wird in Kapitel 1 noch einmal ausführlich dargestellt.

In Kapitel 2 wird aus dem zeitkontinuierlichen Maschinenmodell das zeitdiskrete Maschinenmodell der PM-Synchronmaschine hergeleitet.

In Kapitel 3 werden Methoden des Reglerentwurfs mit der Z-Übertragungsfunktion behandelt. Es wird eine vereinfachte Zustandsreglerentwurfsmethode für die hier vorliegende Strecke mit Blockstruktur dargestellt.

In Kapitel 4 werden Stromreglerstrukturen untersucht und miteinander verglichen.

In Kapitel 5 werden verschiedene Lagereglerstrukturen untersucht. Die Eigenschaften des herkömmlichen Kaskadenreglers, des Zustandsreglers und des kaskadierten Zustandsreglers werden gegenübergestellt.

In Kapitel 6 wird der Entwurf der kaskadierten Zustandsregler zur Lage- und Drehzahlregelung durchgeführt.

In Kapitel 7 wird das PC-Programm zur Simulation der Regelung beschrieben.

In Kapitel 8 wird das auf dem MC 8096 implementierte Regelungsprogramm beschrieben. Dabei werden insbesondere Betrachtungen zur Genauigkeit der verwendeten Algorithmen angestellt.

In Kapitel 9 werden die Ergebnisse der Simulation und die am System gemessenen Ergebnisse zur Strom-, Lage- und Drehzahlregelung dargestellt. In diesem Kapitel findet sich auch eine Übersicht der Kenngrößen der verwendeten Motoren.

1. DIE THEORIE DER PM-SYNCHRONMASCHINE

1.1 VORAUSSETZUNGEN UND BEZEICHNUNGEN

Das Modell der PM-Synchronmaschine läßt sich wie in /4 HETZEL/ aus dem Modell der allgemeinen Synchronmaschine herleiten. Hier soll es direkt aus dem Ersatzschaltbild der PM-Synchronmaschine entwickelt werden. Es wird eine im Stern geschaltete permanentmagnetisch erregten Synchronmaschine ohne Dämpferwicklungen und ohne Nullpunktbelastung zugrundegelegt.

Bei der Bestimmung des Ersatzschaltbildes die folgenden Vereinfachungen gemacht:

- Die durch Inhomogenitäten des Rotors verursachten Abweichungen vom Vollpolverhalten werden vernachlässigt.
- Die Eisenverluste in der Maschine werden vernachlässigt.
- Die Sättigung der magnetischen Kreise wird nicht berücksichtigt.
- Die räumlich verteilten Wicklungen werden durch konzentrierte Wicklungen ersetzt.
- Unsymmetrien durch Fertigungstoleranzen werden vernachlässigt.
- Die Schwächbarkeit des Felds der Permanentmagneten durch die Felder des Stators wird vernachlässigt.

Es werden die folgenden Bezeichnungen verwendet:

R	- Widerstand der Strangwicklung
L_0	- Induktivität der Strangwicklung
M	- Gegeninduktivität zwischen zwei Strängen
ϵ	- elektrischer Rotorwinkel
ω	- elektrische Rotorwinkelgeschwindigkeit
ϵ_{mech}	- mechanischer Rotorwinkel
ω_{mech}	- mechanische Rotorwinkelgeschwindigkeit
z_p	- Polpaarzahl der Maschine

Der Rotorwinkel ϵ wird gemessen zwischen der Achse des Rotors und der Achse des Strangs a .

Um den umständlichen Ausdruck "Winkelgeschwindigkeit" zu vermeiden, wird ω im folgenden auch häufig als Drehzahl bezeichnet.

Wegen des verwendeten Formelzeichens ω sind Verwechslungen nicht zu befürchten.

Es gilt:

$$w = d\varepsilon/dt \quad (1.1.1)$$

$$\varepsilon_{a1} = Z_p * \varepsilon_{mech} \quad (1.1.2)$$

$$w_{mech} = w/Z_p \quad (1.1.3)$$

Die Strangspannungen sind mit u_a , u_b und u_c , die Strangströme mit i_a , i_b und i_c und die in den Strangwicklungen aufgrund der Rotordrehung induzierten Spannungen mit e_a , e_b und e_c bezeichnet. Die induzierten Spannungen werden - wie allgemein üblich - als Spannungsabfälle gedeutet.

1.2 DAS MASCHINENMODELL

Unter den gemachten Voraussetzungen besitzt die PM-Synchronmaschine das in Bild 1.1 dargestellte Ersatzschaltbild.

Aus dem Ersatzschaltbild lassen sich die folgenden Gleichungen /8 NEHL u.a./ ableiten:

$$u_a = R*i_a + L_0 * di_a/dt + M*(di_b/dt + di_c/dt) + e_a \quad (1.2.1a)$$

$$u_b = R*i_b + L_0 * di_b/dt + M*(di_c/dt + di_a/dt) + e_b \quad (1.2.1b)$$

$$u_c = R*i_c + L_0 * di_c/dt + M*(di_a/dt + di_b/dt) + e_c \quad (1.2.1c)$$

Wegen der Sternschaltung gilt:

$$i_a + i_b + i_c = 0 \quad (1.2.2)$$

Unter Verwendung dieser Gleichung erhält man die Spannungsgleichungen:

$$u_a = R*i_a + (L_0 - M) * di_a/dt + e_a \quad (1.2.3a)$$

$$u_b = R*i_b + (L_0 - M) * di_b/dt + e_b \quad (1.2.3b)$$

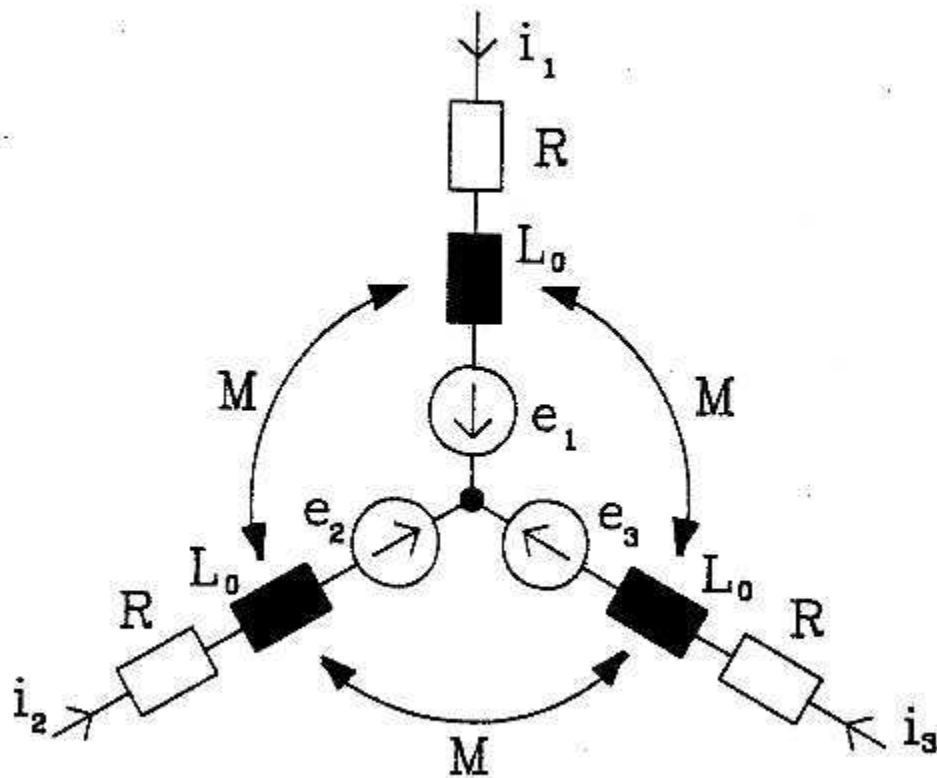


BILD 1.1 Ersatzschaltbild der PM-Synchronmaschine

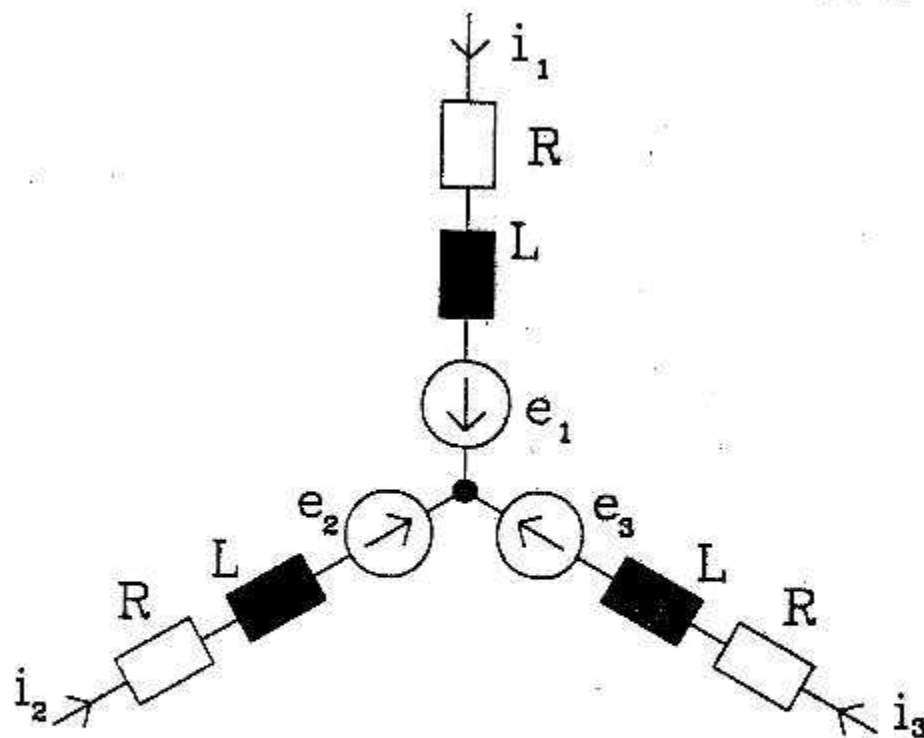


BILD 1.2 Vereinfachtes Ersatzschaltbild
der PM-Synchronmaschine

$$u_c = R \cdot i_c + (L_0 - M) \cdot di_c / dt + e_c \quad (1.2.3c)$$

Die Induktivität

$$L = L_0 - M \quad (1.2.4)$$

soll hier als effektive Stranginduktivität bezeichnet werden.

Die Gleichungen (1.2.3) führen zum in Bild 1.2 dargestellten vereinfachten Ersatzschaltbild der PM-Synchronmaschine.

Für die induzierten Spannungen gilt:

$$\begin{aligned} e_a &= -k_{EMK} \cdot \omega \cdot \sin(\varepsilon) \\ e_b &= -k_{EMK} \cdot \omega \cdot \sin(\varepsilon - 2\pi/3) \\ e_c &= -k_{EMK} \cdot \omega \cdot \sin(\varepsilon - 4\pi/3) \end{aligned} \quad (1.2.5)$$

Dabei ist k_{EMK} die EMK-Konstante.

Wegen der für alle ε gültigen Beziehung:

$$\sin(\varepsilon) + \sin(\varepsilon - 2\pi/3) + \sin(\varepsilon - 4\pi/3) = 0 \quad (1.2.6)$$

folgt:

$$e_a + e_b + e_c = 0 \quad (1.2.7)$$

Aus den Gleichungen (1.2.3) erhält man hiermit die Aussage:

$$u_a + u_b + u_c = 0 \quad (1.2.8)$$

Die Darstellung kann dadurch vereinfacht werden, daß die Stranggrößen jeweils zu einem Vektor zusammengefaßt werden:

$$\begin{aligned} \mathbf{u} &= [u_a \ , \ u_b \ , \ u_c]^T \\ \mathbf{i} &= [i_a \ , \ i_b \ , \ i_c]^T \\ \mathbf{e} &= [e_a \ , \ e_b \ , \ e_c]^T \end{aligned} \quad (1.2.9)$$

Damit erhält man die Spannungsgleichung:

$$u = R \cdot i + L \cdot di/dt + e \quad (1.2.10)$$

1.3 RAUMZEIGERDARSTELLUNG

Die Darstellung der Stranggrößen durch Raumzeiger erfolgt unter Verwendung der komplexen Darstellung.

Hierzu wird zunächst die Konstante $\underline{a}$ definiert durch:

$$\underline{a} = \exp(j \cdot 2\pi/3) \quad (1.3.1)$$

Es gelten dann die Beziehungen:

$$\underline{a} = -1/2 + j \sqrt{3}/2 \quad (1.3.2)$$

$$1 + \underline{a} + \underline{a}^2 = 0 \quad (1.3.3)$$

$$\underline{a}^2 = \underline{a}^{-1} = \underline{a}^* \quad (1.3.4)$$

Für drei gegebene Stranggrößen x_a , x_b und x_c wird die Gültigkeit der Beziehung

$$x_a + x_b + x_c = 0 \quad (1.3.5)$$

vorausgesetzt. Dies gewährleistet die Eindeutigkeit der Darstellung durch Raumzeiger.

Der den Stranggrößen zugeordnete Raumzeiger $\underline{x}$ wird dann definiert durch:

$$\underline{x} = 2/3 \cdot (x_a + x_b \cdot \underline{a} + x_c \cdot \underline{a}^2) \quad (1.3.6)$$

Die Größen x_a , x_b und x_c können durch Projektion des Raumzeigers $\underline{x}$ auf die Achsen 1, $\underline{a}$ und $\underline{a}^2$ gewonnen werden:

$$x_a = \operatorname{Re}(\underline{x}) \quad (1.3.7)$$

$$x_b = \operatorname{Re}(\underline{x} \cdot \underline{a}^{-1}) = -0,5 \cdot \operatorname{Re}(\underline{x}) + \sqrt{3}/2 \cdot \operatorname{Im}(\underline{x}) \quad (1.3.8)$$

$$x_c = \operatorname{Re}(\underline{x} \cdot \underline{a}^{-2}) = -0,5 \cdot \operatorname{Re}(\underline{x}) - \sqrt{3}/2 \cdot \operatorname{Im}(\underline{x}) \quad (1.3.9)$$

Sind umgekehrt die drei Stranggrößen x_a , x_b und x_c gegeben, so lassen sich hieraus die Komponenten des zugeordneten Spannungszeigers nach den folgenden Formeln berechnen:

$$\operatorname{Re}(\underline{x}) = x_a \quad (1.3.10)$$

$$\operatorname{Im}(\underline{x}) = 1/\sqrt{3} \cdot (x_b - x_c) \quad (1.3.11)$$

Unter Verwendung der Beziehung (1.3.5) lassen sich diverse weitere Formeln herleiten. Insbesondere folgt, daß bereits die Kenntnis von zwei Stranggrößen zur Bestimmung des Raumzeigers ausreicht.

Für die Länge $|\underline{x}|$ des Raumzeiger gilt:

$$|\underline{x}|^2 = x_a^2 + 1/3 \cdot (x_b - x_c)^2 \quad (1.3.12)$$

1.4 SPANNUNGSGLEICHUNG DER PM-SYNCHRONMASCHINE IN RAUMZEIGERDARSTELLUNG

Bei Verwendung der Raumzeiger gehen die Spannungsgleichungen (1.1.3) über in die komplexe Spannungsgleichung:

$$\underline{u} = R \cdot \underline{i} + L \cdot d\underline{i}/dt + \underline{e} \quad (1.4.1)$$

Für den Raumzeiger $\underline{e}$ folgt aus den Gleichungen (1.1.5) unter Verwendung von (1.3.10) und (1.3.11) :

$$\underline{e} = -k_{EMK} \cdot \omega \cdot \sin(\varepsilon) - j \cdot 1/\sqrt{3} \cdot \omega \cdot k_{EMK} \cdot (\sin(\varepsilon - 2\pi/3) - \sin(\varepsilon - 4\pi/3))$$

$$\underline{e} = -k_{EMK} \cdot \omega \cdot (\sin(\varepsilon) + j \cdot 1/\sqrt{3} \cdot (\sin(\varepsilon - 2\pi/3) - \sin(\varepsilon - 4\pi/3)))$$

Hieraus erhält man unter Verwendung des Additionstheorems für den Sinus:

$$\underline{e} = -k_{EMK} \cdot \omega \cdot (\sin(\varepsilon) - j \cdot \cos(\varepsilon))$$

$$\underline{e} = j * k_{EMK} * \underline{w} * e^{j\delta} \quad (1.4.2)$$

Der Raumzeiger $\underline{e}$ wird auch als Polradspannungszeiger $\underline{u}_p$ bezeichnet.

Die Einführung von Raumzeigern führt also zu einer sehr einfachen Gleichung für die Polradspannung.

Durch Einsetzen von (1.4.2) in die Spannungsgleichung (1.4.1) erhält man:

$$\underline{u} = R * \underline{i} + L * d\underline{i}/dt + j * k_{EMK} * \underline{w} * e^{j\delta} \quad (1.4.3)$$

1.5 ROTORFESTES KOORDINATENSYSTEM

Werden bei einer linearen komplexen Gleichung oder Differentialgleichung alle auftretenden Größen mit dem Faktor $e^{-j\delta}$ multipliziert, wobei δ eine reelle Zahl ist, so entspricht dies einer Drehung des Koordinatensystems um den Winkel δ .

Die Transformationsgleichung lautet also:

$$\underline{x}^D = \underline{x} * e^{-j\delta} \quad (1.5.1)$$

Der hochgestellte Index D bezeichnet dabei die Größen im gedrehten Koordinatensystem. Die Funktion $\delta = \delta(t)$ wird als differenzierbar vorausgesetzt.

Die Umkehrtransformation ist gegeben durch:

$$\underline{x} = \underline{x}^D * e^{j\delta} \quad (1.5.2)$$

Durch Wahl von

$$\delta(t) = \varepsilon(t) + \varepsilon_0 \quad (1.5.3)$$

in Gleichung (1.5.1) erhält man ein Koordinatensystem, das mit dem Rotor umläuft.

Hierin ist der Winkel ε_0 zunächst noch frei wählbar.

Ein solches Koordinatensystem wird als rotorfestes System bezeichnet.

Ein spezielles rotorfestes System ist das d-q-System. Hierbei ist $\epsilon_0 = 0$ gewählt. Die reelle Achse des Koordinatensystems liegt dann in der Rotorachse.

Im folgenden werden Größen im statorfesten System durch den hochgestellten Index S und Größen im d-q-System durch den hochgestellten Index R gekennzeichnet.

Fehlt diese Indizierung, so geht aus dem Kontext hervor, welches Koordinatensystem zugrundegelegt wird.

Realteil bzw. Imaginärteil einer Größe $\underline{a}^R$ im d-q-System werden mit a_d bzw. a_q bezeichnet:

$$\underline{a}^R = a_d + j \cdot a_q \quad (1.5.4)$$

Die Komponenten einer Größe $\underline{a}^S$ im statorfesten System werden mit a_x und a_y bezeichnet:

$$\underline{a}^S = a_x + j \cdot a_y \quad (1.5.5)$$

1.6 SYSTEMGLEICHUNGEN DER PM-SYNCHRONMASCHINE IM ROTORFESTEN SYSTEM

Mit den Beziehungen:

$$\underline{u}^S = \underline{u}^R \cdot e^{j\epsilon} \quad \underline{i}^S = \underline{i}^R \cdot e^{j\epsilon} \quad \underline{e}^S = \underline{e}^R \cdot e^{j\epsilon} \quad (1.6.1)$$

folgt aus der Spannungsgleichung (1.4.1) im statorfesten System:

$$\underline{u}^R = R \cdot \underline{i}^R + L \cdot d(\underline{i}^R \cdot e^{j\epsilon})/dt \cdot e^{-j\epsilon} + \underline{e}^R$$

$$\underline{u}^R = R \cdot \underline{i}^R + L \cdot d\underline{i}^R/dt + j \cdot d\epsilon/dt \cdot L \cdot \underline{i}^R + \underline{e}^R$$

$$\underline{u}^R = R \cdot \underline{i}^R + L \cdot d\underline{i}^R/dt + j\omega \cdot L \cdot \underline{i}^R + \underline{e}^R$$

Aus (1.4.2) folgt:

$$\underline{u}_p^R = \underline{e}^R = j \cdot k_{EMK} \cdot \omega \quad (1.6.2)$$

Hiermit erhält man die Spannungsgleichung der PM-Synchronmaschine

im d-q-System:

$$\underline{u}^R = R \cdot \underline{i}^R + L \cdot d\underline{i}^R / dt + j\omega \cdot L \cdot \underline{i}^R + j \cdot k_{EMK} \cdot \omega \quad (1.6.3)$$

Durch Zerlegung dieser Gleichung in ihre Komponenten folgt:

$$u_d = R \cdot i_d + L \cdot di_d / dt - \omega \cdot L \cdot i_q \quad (1.6.4)$$

$$u_q = R \cdot i_q + L \cdot di_q / dt + \omega \cdot L \cdot i_d + k_{EMK} \cdot \omega \quad (1.6.5)$$

Der Momentanwert P der im Stator umgesetzten elektrischen Leistung beträgt /1 BÜHLER/:

$$P = 3/2 \cdot \operatorname{Re}(\underline{u}^* \cdot \underline{i}) \quad (1.6.6)$$

Die Verlustleistung P_v im Stator ist gegeben durch:

$$P_v = 3/2 \cdot R \cdot i^2 \quad (1.6.7)$$

Der Momentanwert m_{e1} des elektrisch erzeugten Drehmoments beträgt:

$$m_{e1} = 3/2 \cdot z_p \cdot k_{EMK} \cdot i_q \quad (1.6.8)$$

Mit der Momentenkonstante k_{MOM} :

$$k_{MOM} = 3/2 \cdot z_p \cdot k_{EMK} \quad (1.6.9)$$

gilt:

$$m_{e1} = k_{MOM} \cdot i_q \quad (1.6.10)$$

Die mechanische Gleichung des Systems lautet:

$$J \cdot d\omega / dt = m_{e1} - k_R \cdot \omega - m_l \quad (1.6.11)$$

Dabei ist J das Trägheitsmoment von Rotor und Last, k_R der Reibungskoeffizient und m_l das Lastmoment.

Bild 1.3 zeigt das Strukturbild der PM-Synchronmaschine im rotorfesten System.

1.7 SYSTEMGLEICHUNGEN DER PM-SYNCHRONMASCHINE IM STATORFESTEN SYSTEM

Die Spannungsgleichung im statorfesten System lautet:

$$\underline{u}^s = R \cdot \underline{i}^s + L \cdot d\underline{i}^s / dt + j \cdot k_{EMK} \cdot \omega \cdot e^{j\theta} \quad (1.7.1)$$

Durch Zerlegung der Spannungsgleichung in ihre Komponenten erhält man:

$$u_x = R \cdot i_x + L \cdot di_x / dt + k_{EMK} \cdot \omega \cdot e^{j\theta} \quad (1.7.2)$$

$$u_y = R \cdot i_y + L \cdot di_y / dt + k_{EMK} \cdot \omega \cdot e^{j\theta} \quad (1.7.3)$$

Die Momentengleichung nimmt im statorfesten System die folgende Gestalt an:

$$m_e = 3/2 \cdot z_p \cdot k_{EMK} \cdot \text{Im}(\underline{i}^s \cdot e^{-j\theta}) \quad (1.7.4)$$

Die mechanische Gleichung (1.6.11) bleibt unverändert.

Bild 1.4 zeigt das Strukturbild der PM-Synchronmaschine im statorfesten System.

1.8 BEZOGENES GRÖßENSYSTEM

Bei der hier durchgeführten Einführung bezogener Größen werden die Bezugsgrößen so gewählt, daß die Maschinengleichungen eine möglichst einfache Form annehmen.

Im folgenden werden zunächst die Bezugsgrößen definiert und anschließend die Maschinengleichungen in bezogenen Größen abge-

leitet. Als Bezugsgrößen werden die folgenden Größen verwendet:

$$T_{e1} = L/R \quad (1.8.1)$$

$$\omega_0 = R/L = 1/T_{e1} \quad (1.8.2)$$

$$U_0 = k_{EMK} * \omega_0 \quad (1.8.3)$$

$$I_0 = k_{EMK} / L \quad (1.8.4)$$

Bei der Winkelgeschwindigkeit ω_0 herrscht zwischen Spannung und Strom in der Maschine eine Phasendifferenz von 45 Grad. Deshalb wird sie auch als 45°-Winkelgeschwindigkeit bezeichnet.

U_0 ist der Betrag der Spannung, die vom Rotor im Stator induziert wird, wenn die Maschine mit der Winkelgeschwindigkeit ω_0 betrieben wird.

Aus der Darstellung $I_0 = k_{EMK} * \omega_0 / (L \omega_0)$ folgt, daß I_0 der Betrag des Stroms ist, der bei der Winkelgeschwindigkeit $\omega = \infty$ im Stator fließt, wenn die Maschine im Kurzschlußbetrieb betrieben wird.

Für den Betrag I_{45} des Stroms, der bei $\omega = \omega_0$ im Kurzschlußbetrieb fließt, gilt: $I_{45} = I_0 / \sqrt{2}$.

Aus den Größen ω , $\underline{u}$, $\underline{i}$ und t werden nun die entsprechenden bezogenen Größen gebildet. Die bezogenen Größen werden dabei durch Apostrophe von den ursprünglichen Größen unterschieden:

$$t' = t/T_{e1} = t/(L/R) \quad (1.8.5)$$

$$\omega' = \omega/\omega_0 = \omega * L/R \quad (1.8.6)$$

$$\underline{u}' = \underline{u}/U_0 = \underline{u}/(k_{EMK} * R/L) \quad (1.8.7)$$

$$\underline{i}' = \underline{i}/I_0 = \underline{i}/(k_{EMK} / L) \quad (1.8.8)$$

1.9 UMRECHNUNG DER MASCHINENGLEICHUNGEN IN BEZOGENE GRÖßEN

Durch Auflösung der Definitionsgleichungen für die bezogenen Größen nach den ursprünglichen Größen erhält man:

$$t = t' \cdot T_{e1} = t' \cdot (L/R) \quad (1.9.1)$$

$$w = w' \cdot w_0 = w' \cdot (L/R) \quad (1.9.2)$$

$$\underline{u} = \underline{u}' \cdot U_0 = \underline{u}' \cdot (k_{EMK} \cdot R/L) \quad (1.9.3)$$

$$\underline{i} = \underline{i}' / I_0 = \underline{i}' \cdot (k_{EMK} / L) \quad (1.9.4)$$

Durch Einsetzen in die Spannungsgleichung im rotorfesten System (1.6.3) folgt:

$$U_0 \cdot \underline{u}' = R \cdot I_0 \cdot \underline{i}' + L \cdot I_0 \cdot d\underline{i}' / dt + jw \cdot L \cdot I_0 \cdot \underline{i}' + jw_0 \cdot w' \cdot k_{EMK}$$

Daraus erhält man die Spannungsgleichung in bezogenen Größen:

$$\underline{u}' = \underline{i}' + d\underline{i}' / dt' + jw' \cdot \underline{i}' + jw' \quad (1.9.5)$$

Aus der Momentengleichung (1.6.8) folgt mit (1.9.4):

$$m_{e1} = 3/2 \cdot z_p \cdot k_{EMK}^2 / L \cdot i_q' \quad (1.9.6)$$

Es wird nun die bezogene Momentenkonstante k_{MOM}' eingeführt:

$$k_{MOM}' = 3/2 \cdot z_p \cdot k_{EMK}^2 / L = k_{MOM} \cdot k_{EMK} / L \quad (1.9.7)$$

Die bezogene Momentenkonstante hat die Dimension des Drehmoments. Es gilt dann:

$$\hat{m}_{e1} = k_{MOM}' \cdot i_q' \quad (1.9.8)$$

Aus der mechanischen Gleichung (1.6.11) erhält man durch Einset-

zen von (1.9.1) und (1.9.2):

$$J * d(w_0 * w') / d(t' * T_{e1}) = m_{e1} - k_R * w_0 * w' - m_1$$

$$J / T_{e1}^2 * dw' / dt' = m_{e1} - k_R * w_0 * w' - m_1 \quad (1.9.9)$$

Das bezogene Trägheitsmoment J' wird definiert durch:

$$J' = J / T_{e1}^2 \quad (1.9.10)$$

Der bezogene Reibungskoeffizient k_R' wird definiert durch:

$$k_R' = k_R * w_0 \quad (1.9.11)$$

Die Konstanten J' und k_R' haben wie bereits die Konstante k_{mom}' die Dimension des Drehmoments.

Es gilt dann:

$$J' * dw' / dt' = m_{e1} - k_R' * w' - m_1 \quad (1.9.12)$$

Bei der Darstellung eines gegebenen Systems in bezogenen Größen bestimmen die folgenden Parameter das Verhalten :

1. Die zur Verfügung stehende Spannung $u_{max}' = u_{max} / U_0$
Die zur Verfügung stehende Spannung u_{max} wird bestimmt durch die verwendete Zwischenkreisspannung.
2. Der zulässige Statorstrom $i_{max}' = i_{max} / I_0$.
Der zulässige Strom i_{max}' wird bestimmt durch die aufgrund der ohmschen Verluste auftretende Erwärmung der Maschine.
Zu unterscheiden ist hier noch zwischen dem kurzzeitig zulässigen Strom $i_{max,k}'$ und dem zulässigen Dauerstrom $i_{max,p}'$.
3. Die bezogene Momentenkonstante k_{mom}'
4. Das bezogene Trägheitsmoment J'

Ferner ist noch die maximale Winkelgeschwindigkeit w_{max}' , bis zu der der Motor betrieben werden soll, von Bedeutung.

Bild 1.5 zeigt das Strukturbild der PM-Synchronmaschine im rotorfesten System bei Verwendung der bezogenen Größen.

In den folgenden Kapiteln wird bei der Untersuchung des Systems stets das bezogene Größensystem benutzt. Die zur Unterscheidung der bezogenen Größen von den absoluten Größen verwendeten Apostrophe werden weggelassen.

1.10 LINEARISIERUNG DES MASCHINENMODELLS

Zur besseren Übersicht sind hier die Gleichungen zusammengestellt, die das Verhalten der Maschine bei Verwendung bezogener Größen im rotorfesten System beschreiben. Die zur Kennzeichnung bezogener Größen verwendeten Apostrophe sind im folgenden weggelassen.

$$\underline{u} = \underline{i} + \frac{d\underline{i}}{dt} + j\omega \underline{i} + j\omega \quad (1.10.1)$$

$$m_{e1} = k_{M0M} \cdot i_q \quad (1.10.2)$$

$$J \cdot \frac{dw}{dt} = m_{e1} - k_R \cdot w - m_1 \quad (1.10.3)$$

Diese Gleichungen bilden ein nichtlineares Differentialgleichungssystem 4. Ordnung. Während auf den elektrischen Teil des Systems Störgrößen nur in sehr geringem Umfang wirken, ist der mechanische Teil des Systems Störgrößen in hohem Maße ausgesetzt.

In Matrixschreibweise kann das Differentialgleichungssystem wie folgt dargestellt werden:

$$\frac{d}{dt} \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -1 & w & 0 & 0 \\ -w & -1 & -1 & 0 \\ 0 & k_{M0M}/J & -k_R/J & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} + \begin{bmatrix} u_d \\ u_q \\ -m_1/J \\ 0 \end{bmatrix} \quad (1.10.4)$$

Die Terme des letzten Vektors werden dabei als konstant betrachtet. Dies bedeutet, daß sowohl die Eingangsgrößen u_d und u_q als

auch die Größen m_1 und J als konstant betrachtet werden.

Es liegt nun nahe, eine geschlossene Lösung des obigen Differentialgleichungssystems für ein kleines Zeitintervall dadurch zu erzielen, daß man bestimmte Terme, die eine Nichtlinearität verursachen, als konstant betrachtet.

Man muß sich jedoch der Tatsache bewußt sein, daß die dann erzielte Lösung nur eine Näherungslösung sein wird.

Die Tatsache, daß eine bestimmte Größen als konstant betrachtet wird, wird im folgenden durch den tiefgestellten Index 1 gekennzeichnet.

Betrachtet man die Winkelgeschwindigkeit w in der Spannungsgleichung (1.10.1) als konstant, so erhält man das lineare DGL-System:

$$\frac{d}{dt} \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -1 & w_1 & 0 & 0 \\ -w_1 & -1 & 0 & 0 \\ 0 & k_{M0M}/J & -kr/J & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} * \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} + \begin{bmatrix} u_d \\ u_q - w_1 \\ -m_1/J \\ 0 \end{bmatrix} \quad (1.10.5)$$

Dieses DGL-System hat den Nachteil, daß darin die darin auftretenden Koeffizienten w_1 zustands- und damit zeitabhängig sind. Die Übergangsfunktion des Stroms i unter der Annahme konstanter Winkelgeschwindigkeit $w = w_1$ ist in /4 HETZEL/ abgeleitet worden. Aus den dort abgeleitete Gleichungen (5.2.1) bis (5.2.3) erhält man die Gleichung:

$$\underline{i}(t) = \underline{i}(0) * e^{-(t+jw_1t)} + (\underline{u} - jw)/(1+jw) * (1 - e^{-(t+jw_1t)}) \quad (1.10.6)$$

Diese Gleichung zeigt, daß das Übergangsverhalten in komplizierter Weise von der Drehzahl abhängig ist.

Ein DGL-System mit konstanten Koeffizienten erhält man dadurch, daß die in (1.10.4) auftretenden Terme $w*i_q$ und $w*i_d$ als konstant

angesehen werden. Dies führt zu dem DGL-System:

$$\frac{d}{dt} \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & k_{M0M}/J & -k_r/J & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} * \begin{bmatrix} i_d \\ i_q \\ w \\ \varepsilon \end{bmatrix} + \begin{bmatrix} u_d + w_1 * i_{q1} \\ u_q - w_1 * i_{d1} - w_1 \\ -m_1/J \\ 0 \end{bmatrix} \quad (1.10.7)$$

Dieses DGL-System dient im folgenden Kapitel als Ausgangspunkt für die Entwicklung eines zeitdiskreten Maschinenmodells im rotorfesten System.

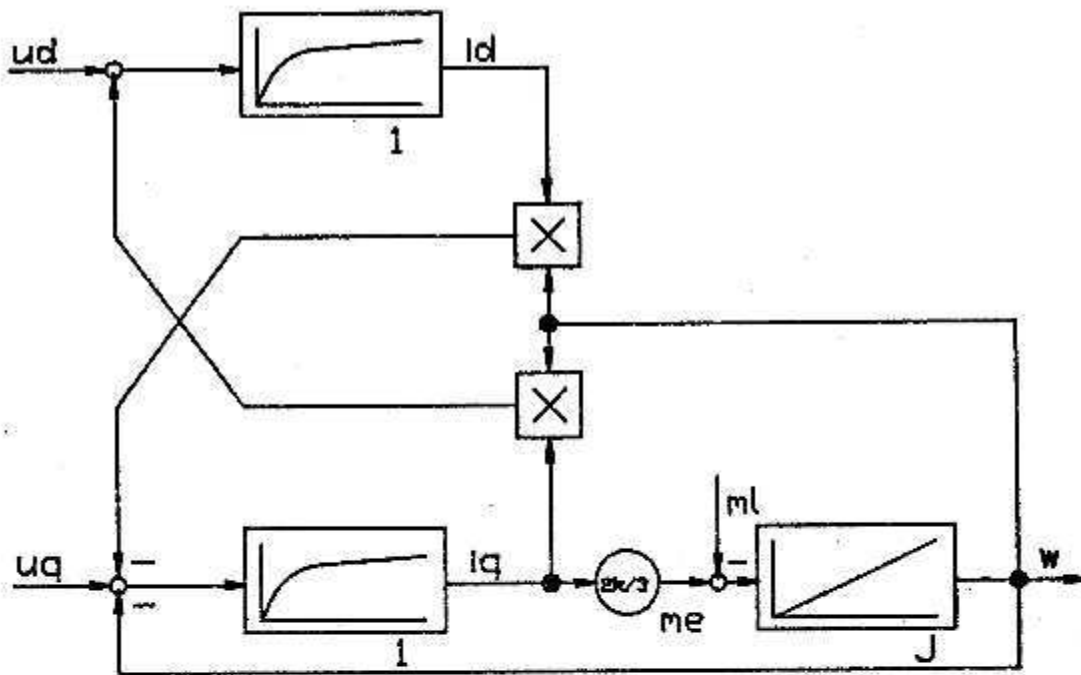


BILD 1.5 Strukturbild der PM-Synchronmaschine
im rotorfesten System in bezogenen Größen

2. ZEITDISKRETES MASCHINENMODELL

2.1 HERLEITUNGSVERFAHREN

Die Herleitung eines zeitdiskreten Übergangsmodells aus einem zeitkontinuierlichen Modell erfolgt nach /3 HARTMANN/ .

Ausgangspunkt ist das DGL-System:

$$\frac{dx}{dt} = A * x + b(t) \quad (2.1.1)$$

Darin ist x der Zustandsvektor , A eine konstante Matrix und b ein Vektor, der die Eingangsgrößen und konstante Störgrößen enthält.

Die allgemeine Lösung dieses DGL-Systems ist - wie aus der Theorie der Differentialgleichungen bekannt ist - gegeben durch:

$$x(t) = \exp(A*(t-t_0)) * x(t_0) + \int_{t_0}^t \exp(A*(t-\tau)) * b(\tau) d\tau \quad (2.1.2)$$

Führt man hier die Transitionsmatrix:

$$\Phi(t) = \exp(A*t) \quad (2.1.3)$$

ein, so folgt:

$$x(t) = \Phi(t-t_0) * x(t_0) + \int_{t_0}^t \Phi(t-\tau) * b(\tau) d\tau \quad (2.1.4)$$

Es wird nun der Fall betrachtet, daß der Vektor b innerhalb des Intervalls $[t_0, t_0+T]$ konstant ist.

Dann gilt:

$$x(t_0+T) = \Phi(T) * x(t_0) + \int_{t_0}^{t_0+T} \Phi(T-\tau) * b d\tau$$

Durch Substitution vereinfacht sich diese Gleichung zu:

$$x(t_0+T) = \Phi(T) * x(t_0) + \int_0^T \Phi(t) dt * b \quad (2.1.5)$$

Mit dem Eingangsvektor H :

$$H(T) = \int_0^T \Phi(t) dt * b \quad (2.1.6)$$

erhält man das zeitdiskrete Übergangsmodell:

$$x(t_0+T) = \Phi(T) * x(t_0) + H(T) \quad (2.1.7)$$

Der Einfluß von Störgrößen läßt sich durch zusätzlichen Störgrößeneingangsvektor D(T) berücksichtigen:

$$x(t_0+T) = \Phi(T) * x(t_0) + H(T) + D(T) \quad (2.1.8)$$

Die Bestimmung der Transitionsmatrix und des Eingangsvektors kann nach /3 HARTMANN, S.4.18/ erfolgen.

Es gilt:

$$\Phi(T) = \sum_{k=0}^{\infty} A^k * T^k / k! \quad (2.1.9)$$

$$H(T) = \sum_{k=0}^{\infty} \frac{T^{k+1} * A^k * b}{(k+1) * k!} \quad (2.1.10)$$

Eine Möglichkeit der Bestimmung von $\Phi(T)$ unter Verwendung der Laplace-Transformation liefert die Gleichung:

$$\mathcal{L}(\Phi(T)) = (sE - A)^{-1} \quad (2.1.11)$$

2.2 ZEITDISKRETES MASCHINENMODELL MIT BERÜCKSICHTIGUNG DER REIBUNG

Die Herleitung des zeitdiskreten Maschinenmodells erfolgt hier nicht nach den Gleichungen (2.1.9) und (2.1.10) , sondern durch direkte Lösung des DGl-Systems.

Ausgegangen wird vom Gleichungssystem (1.10.7) im rotorfesten System. Die Gleichungen des Systems lauten:

$$di_d/dt = -i_d + u_d + w_1 * i_{q1} \quad (2.2.1)$$

$$di_q/dt = -i_q + u_q - w_1 * i_{q1} - w_1 \quad (2.2.2)$$

$$dw/dt = k_{MOM}/J * i_q - k_R/J * w - m_1/J \quad (2.2.3)$$

$$d\varepsilon/dt = w \quad (2.2.4)$$

Mit der kompensierten Spannung u_k :

$$u_k = u - jw_1 * i_{q1} - w_1 \quad (2.2.5)$$

erhält man:

$$i_d(t) = i_d(0) * e^{-t} + u_{k,d} * (1 - e^{-t}) \quad (2.2.6)$$

$$i_q(t) = i_q(0) * e^{-t} + u_{k,q} * (1 - e^{-t}) \quad (2.2.7)$$

Durch Einsetzen von (2.2.7) in Gleichung (2.2.3) folgt:

$$dw/dt = k_{MOM}/J * (i_q(0) * e^{-t} + u_{k,q} * (1 - e^{-t})) - k_R/J * w - m_1/J$$

$$dw/dt = k_{MOM}/J * (i_q(0) - u_{k,q}) * e^{-t} + k_{MOM}/J * u_{k,q} - k_R/J * w - m_1/J \quad (2.2.8)$$

Mit den Abkürzungen:

$$A = k_{MOM}/J * (i_q(0) - u_{k,q}) \quad B = k_{MOM}/J * u_{k,q} - m_1/J \quad C = k_R/J$$

erhält man aus obiger DGL:

$$dw/dt = A * e^{-t} + B - C * w \quad (2.2.9)$$

Die Lösung dieser DLG lautet:

$$w(t) = B/C + A/(C-1) * e^{-t} + d * e^{-Ct} \quad (2.2.10)$$

Dabei ist d eine Konstante, die aus der Anfangsdrehzahl $w(0)$ be-

stimmt werden kann.

Es gilt:

$$w(0) = B/C + A/(C-1) + d$$

Damit folgt aus (2.2.10):

$$w(t) = B/C + A/(C-1) * e^{-t} + (w(0) - B/C - A/(C-1)) * e^{-ct} \quad (2.2.11)$$

Mit den Konstanten:

$$a = k_{MOM}/k_R \quad b = k_{MOM}/(k_R - J) \quad d = 1/(k_R - J)$$

folgt:

$$B/C = a * u_{k,q} - m_1/k_R$$

und

$$A/(C-1) = b * i_q(0) - d * u_{k,q}$$

Durch Einsetzen in (2.2.11) erhält man:

$$\begin{aligned} w(t) &= a * u_{k,q} - m_1/k_R + (b * i_q(0) - d * u_{k,q}) * e^{-t} + w(0) * e^{-ct} - \\ &\quad - (a * u_{k,q} - m_1/k_R + b * i_q(0) - d * u_{k,q}) * e^{-ct} \\ w(t) &= e^{-ct} * w(0) + (e^{-t} - e^{-ct}) * b * i_q(0) + \\ &\quad + (a - d * e^{-t} - (a-d) * e^{-ct}) * u_{k,q} - (1 - e^{-ct})/k_R * m_1 \end{aligned} \quad (2.2.12)$$

Damit ist die Übergangsfunktion für die Drehzahl bestimmt worden. Durch Integration dieser Funktion erhält man die Übergangsfunktion für die Lage $\varepsilon(t)$:

$$\begin{aligned} \varepsilon(t) &= \varepsilon(0) + 1/C * (1 - e^{-ct}) * w(0) + (1 - e^{-t} + 1/C * (e^{-ct} - 1)) * b * i_q(0) + \\ &\quad + (at + d * (e^{-t} - 1) + (a-d)/C * (e^{-ct} - 1)) * u_{k,q} - \\ &\quad - (t + 1/C * (e^{-ct} - 1))/k_R * m_1 \end{aligned} \quad (2.2.13)$$

Es wird nun der Zustandsvektor:

$$\mathbf{x} = [i_q, w, \varepsilon] \quad (2.2.14)$$

zugrundegelegt. Eingangsgröße ist $u_{k,q}$.

Die Elemente der Transitionmatrix $\Phi(T)$ seien mit a_{ik} bezeichnet. Für die nicht verschwindenden Elemente von $\Phi(T)$ gilt dann:

$$\begin{aligned} a_{11} &= e^{-T} & a_{21} &= (e^{-T} - e^{-CT}) * b \\ a_{22} &= e^{-CT} & a_{31} &= (1 - e^{-T} + 1/C * (e^{-CT} - 1)) * b \\ a_{32} &= 1/C * (1 - e^{-CT}) & a_{33} &= 1 \end{aligned} \quad (2.2.15)$$

Die Komponenten des Eingangsvektors $H(T)$ seien mit h_i bezeichnet. Für die Komponenten von $H(T)$ gilt:

$$\begin{aligned} h_1 &= 1 - e^{-T} \\ h_2 &= a - d * e^{-T} - (a - d) * e^{-CT} \\ h_3 &= aT + d * (e^{-T} - 1) + (a - d) / C * (e^{-CT} - 1) \end{aligned} \quad (2.2.16)$$

Die Komponenten des Störgrößeneingangsvektors $D(T)$ seien mit d_i bezeichnet.

Für die Komponenten von $D(T)$ gilt:

$$\begin{aligned} d_1 &= 0 & d_2 &= - (1 - e^{-CT}) / k_R & d_3 &= - (T + 1/C * (e^{-CT} - 1)) / k_R \end{aligned} \quad (2.2.17)$$

Ein Übergangsmodell für den mechanischen Teil des Systems entsteht dadurch, daß in Gleichung (2.2.3) i_q als konstant behandelt wird.

Mit den oben benutzten Abkürzungen ist die Lösung der DLG:

$$dw/dt = k_{MOM} / J * i_q - k_R / J * w$$

gegeben durch:

$$w(t) = e^{-ct} * w(0) + (1 - e^{-ct}) * a * i_q \quad (2.2.18)$$

Durch Integration erhält man hieraus:

$$\varepsilon(t) = \varepsilon(0) + 1/C * (1 - e^{-ct}) * w(0) + (t + 1/C * (e^{-ct} - 1)) * a * i_q \quad (2.2.19)$$

Mit dem Zustandsvektor $x = [w, \varepsilon]$ gilt dann:

$$x_{n+1} = \begin{bmatrix} a_{22} & 0 \\ a_{32} & 1 \end{bmatrix} * x + \begin{bmatrix} (1 - e^{-cT}) * a \\ (T + 1/C * (e^{-cT} - 1)) * a \end{bmatrix} * i_q \quad (2.2.20)$$

2.3 ZEITDISKRETES MASCHINENMODELL BEI VERNACHLÄSSIGUNG DER REIBUNG

Die Übergangsfunktion im Fall $k_R = 0$ kann durch den Grenzübergang $k_R \rightarrow 0$ aus Gleichung (2.2.12) gewonnen werden.

Einfacher ist es jedoch, die DGL für diesen Fall zu lösen.

Sei definiert:

$$d = k_{MOM} / J$$

Mit $k_R = 0$ folgt dann aus (2.2.8):

$$dw/dt = d * (i_q(0) - u_{K,q}) * e^{-t} + d * u_{K,q} - m_I / J \quad (2.3.1)$$

Die Lösung dieser DGL erhält man durch Integration:

$$w(t) = w(0) + d * (1 - e^{-t}) * i_q(0) + d * (e^{-t} + t - 1) * u_{K,q} - t/J * m_I \quad (2.3.2)$$

Nochmalige Integration dieser Gleichung liefert:

$$\begin{aligned} \varepsilon(t) = & \varepsilon(0) + t \cdot w(0) + d \cdot (t + e^{-t} - 1) \cdot i_q(0) + \\ & + d \cdot (-e^{-t} + 1 + t^2/2 - t) \cdot u_{k,q} - t^2/(2 \cdot J) \cdot m_i \end{aligned} \quad (2.3.3)$$

Mit dem Zustandsvektor nach (2.2.14) gilt für die nicht verschwindenden Elemente von $\Phi(T)$:

$$\begin{aligned} a_{11} &= e^{-T} & a_{21} &= d \cdot (1 - e^{-T}) \\ a_{22} &= 1 & a_{31} &= d \cdot (T + e^{-T} - 1) \\ a_{32} &= T & a_{33} &= 1 \end{aligned} \quad (2.3.4)$$

Für die Komponenten von $H(T)$ gilt:

$$\begin{aligned} h_1 &= 1 - e^{-T} \\ h_2 &= d \cdot (e^{-T} + T - 1) \\ h_3 &= d \cdot (-e^{-T} + 1 + T^2/2 - T) \end{aligned} \quad (2.3.5)$$

Für die Komponenten von $D(T)$ gilt:

$$\begin{aligned} d_1 &= 0 \\ d_2 &= -T/J \\ d_3 &= -T^2/(2 \cdot J) \end{aligned} \quad (2.3.6)$$

Im folgenden soll anhand eines Beispiels gezeigt werden, daß sich den im zeitdiskreten Übergangsmodell auftretenden Konstanten eine physikalische Bedeutung zuordnen läßt.

Für das mittlere Drehmoment m_{e1} erhält man:

$$\begin{aligned} T \cdot m_{e1} &= \int_0^T m_{e1}(t) \, dt = \int_0^T k_{MOM} \cdot i_q(t) \, dt = \\ &= k_{MOM} \cdot (1 - e^{-T}) \cdot i_q(0) + k_{MOM} \cdot (e^{-T} + T - 1) \cdot u_{k,q} = \end{aligned}$$

$$= J * (a_{32} * i_q(0) + h_3 * u_{k,q})$$

(2.3.7)

Im mechanischen Teil des Übergangsmodells wird also das mittlere Drehmoment $m_{e1}(t)$ zur Bestimmung der Drehzahl verwendet.

3. Z-TRANSFORMATION UND REGLERENTWURF

3.1 DIE Z-TRANSFORMATION

Ist (f_n) eine Folge, so wird die Z-Transformierte $F(z)$ der Folge definiert durch:

$$F(z) = \sum_{n=0}^{\infty} f_n z^{-n} \quad (3.1.1)$$

Die Abbildung

$$\mathcal{Z}: (f_n) \rightarrow F(z) \quad (3.1.2)$$

wird als Z-Transformation bezeichnet.

Die Potenzreihe (3.1.1) konvergiert gleichmäßig absolut für alle z mit :

$$|z| > R = \lim_{n \rightarrow \infty} \sup \sqrt[n]{|f_n|} \quad (3.1.3)$$

Der Wert R wird als Konvergenzradius der Potenzreihe bezeichnet. Für jede beschränkte Folge existiert somit ein Konvergenzradius $R < \infty$. Bei bestimmten unbeschränkten Folgen gilt jedoch $R = \infty$. In diesem Fall existiert die Z-Transformierte der Folge nicht.

3.2 EIGENSCHAFTEN DER Z-TRANSFORMATION

Im Folgenden wird die Existenz der Z-Transformierten vorausgesetzt. Die gemachten Aussagen sind gültig im gemeinsamen Konvergenzbereich der Z-Transformierten.

Die Z-Transformation besitzt folgende Eigenschaften:

1. Linearitätssatz:

$$((c_1 * f_n + c_2 * g_n)) = c_1 * ((f_n)) + c_2 * ((g_n)) \quad (3.2.1)$$

2. Verschiebungssatz 1:

$$((f_{n-k})) = z^{-k} * ((f_n)) \quad (3.2.2)$$

3. Verschiebungssatz 2:

$$((f_{n+k})) = z^k * ((f_n)) - A \quad (3.2.3)$$

, wobei gilt:

$$A = \sum_{n=0}^{k-1} f_n z^{-n}$$

4. Dämpfungssatz:

$$((c^n f_n)) = F(z/c) \quad (3.2.4)$$

5. Faltungssatz:

Für die Folge (h_n) mit:

$$h_n = \sum_{k=0}^n f_{n-k} * g_k$$

gilt:

$$H(z) = F(z) * G(z) \quad (3.2.5)$$

6. Anfangswertsatz:

$$\lim_{z \rightarrow \infty} F(z) = f_0 \quad (3.2.6)$$

7. Endwertsatz:

Existiert

$$a = \lim_{n \rightarrow \infty} f_n$$

, ist $R \leq 1$ und ist $F(z)$ eine gebrochen rationale Funktion
, so gilt:

$$\lim_{z \rightarrow 1+0} ((z-1) * F(z)) = a \quad (3.2.7)$$

Im folgenden sind die Z-Transformierte einfacher Folgen angegeben.

Die Sprungfolge $f_n = 1$ ($n=0,1,\dots$) hat die Z-Transformierte:

$$F(z) = z/(z-1)$$

Die Rampenfolge $f_n = nT$ ($n=0,1,\dots$) hat die Z-Transformierte:

$$F(z) = Tz/(z-1)^2$$

Die Folge $f_n = (nT)^2$ ($n=0,1,\dots$) hat die Z-Transformierte:

$$F(z) = T^2 z(z+1)/(z-1)^3$$

Die Potenzfolge $f_n = a^n$ ($n=0,1,\dots$) hat die Z-Transformierte:

$$F(z) = z/(z-a)$$

Die Folge $f_n = nT * a^n$ ($n=0,1,\dots$) hat die Z-Transformierte:

$$F(z) = Taz/(z-a)^2$$

3.3 INVERSE Z-TRANSFORMATION

Die inverse Z-Transformation $\mathcal{Z}^{-1}$ bildet die Funktion $F(z)$ auf die Folge (f_n) ab.

Die Berechnung erfolgt unter Benutzung des Cauchy'schen Integralsatzes. Bei rationalen Funktionen mit den Polstellen $p_1, p_2, \dots, p_k$ gilt:

$$f_n = \sum_{k=1}^K \text{Res}(F(z) * z^{n-1}, p_k) \quad (3.3.1)$$

Ist

$$F(z) = \frac{a}{z-p} \quad (3.3.2)$$

, so gilt:

$$f_n = \text{Res}(a * z^{n-1} / (z-p), p) = a * p^{n-1} \quad (3.3.3)$$

Ist

$$F(z) = \frac{a}{(z-p)^k} \quad (3.3.4)$$

, so gilt:

$$f_n = \frac{a}{(k-1)!} * \left. \frac{d^{k-1} z^{n-1}}{dz^{k-1}} \right|_{z=p} \quad (3.3.5)$$

Die Berücksichtigung eines Zählerpolynoms in (3.3.4) kann durch Verwendung des Verschiebungssatzes erfolgen.

Nach Durchführung einer Partialbruchzerlegung läßt sich eine gegebene rationale Funktion $F(z)$ unter Verwendung des Linearitätssatzes in den Folgenbereich transformieren.

3.4 EIGENSCHAFTEN DER Z-ÖF DIGITALER REGLER

Das Reglergesetz eines zeitdiskret arbeitenden linearen Reglers mit einer Eingangsgröße x und einer Ausgangsgröße y hat die Form:

$$y_n = - \sum_{k=1}^N a_k * y_{n-k} + \sum_{k=0}^M b_k * x_{n-k} \quad (3.4.1)$$

Siehe hierzu /10 SCHÖNFELD 1.4 S.21/ und /3 HARTMANN, S.4.56/.

Der Regler ist also ein digitales Filter.

Das Auftreten der Summanden $a_k * y_{n-k}$ ($k=1..N$) kennzeichnet das Filter als rekursives Filter.

Gilt $a_k = 0$ für $k=1..N$, so entsteht ein Transversalfilter.

Durch Z-Transformation unter Verwendung des Verschiebungssatzes folgt:

$$y(z) = - \sum_{k=1}^N a_k z^{-k} * y(z) + \sum_{k=0}^M b_k z^{-k} * x(z) \quad (3.4.2)$$

Die Z-Übertragungsfunktion $G(z)$ des digitalen Filters ist definiert durch:

$$G(z) = \frac{y(z)}{x(z)} \quad (3.4.3)$$

Bei einem Regler wird die Übertragungsfunktion (ÖF) auch als Führungs-ÖF $G_f(z)$ bezeichnet.

In den hier betrachteten Fällen ist $G(z)$ stets eine gebrochen rationale Funktion. Im folgenden bezeichnet $N(z)$ das Nennerpolynom und $Z(z)$ das Zählerpolynom einer ÖF $G(z)$.

Aus (3.4.3) folgt mit $a_0 = 1$:

$$G(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{\sum_{k=0}^N a_k z^{-k}} \quad (3.4.4)$$

Nach /10 SCHÖNFELD S.22/ kann nun eine Hilfsvariable $v(z)$ rekur-

siv definiert werden durch:

$$v(z) = x(z) - (a_1 * z^{-1} + \dots + a_N * z^{-N}) * v(z) \quad (3.4.5)$$

Im Unterschied zu Gleichung (3.4.2) enthält (3.4.5) nur den gegenwärtigen Wert der Eingangsgröße.

Nach der Umformung:

$$v(z) = x(z) / (1 + a_1 * z^{-1} + \dots + a_N * z^{-N})$$

folgt durch Einsetzen in (3.4.2):

$$y(z) = \sum_{k=0}^M b_k * z^{-k} * v(z) \quad (3.4.6)$$

Rücktransformation in den Folgenbereich liefert:

$$v_n = x_n - (a_1 v_{n-1} + \dots + a_N v_{n-N}) \quad (3.4.7a)$$

$$y_n = b_0 v_n + b_1 v_{n-1} + \dots + b_M v_{n-M} \quad (3.4.7b)$$

Das rekursive Filter kann also unter Verwendung der Hilfsvariablen v_n in einen durch (3.4.7a) beschriebenen rekursiven Teil und einen durch (3.4.7b) beschriebenen nachfolgenden transversalen Teil zerlegt werden.

Der zur Berechnung der Ausgangsgröße notwendige Aufwand ist für die beiden durch (3.4.1) und (3.4.7) gegebenen Formen des Reglergesetzes gleich. Die zweite Form benötigt jedoch weniger Speicherplatz, da hier nur die Werte $v_n, \dots, v_{n-N}$ zu speichern sind.

Die hier sehr allgemeinen Ausführungen können am Beispiel eines digitalen PI-Reglers konkretisiert werden.

Das Reglergesetz lautet in der zweiten Form:

$$I_n = I_{n-1} + x_n \quad (3.4.8)$$

$$y_n = k_1 * I_n + k_2 * x_n$$

Die Umwandlung in die erste Form liefert:

$$y_n = y_{n-1} + (k_1 + k_2) * x_n - k_2 * x_{n-1} \quad (3.4.9)$$

Es sollen nun die Übertragungsfunktionen realisierbarer Filter (Regler) untersucht werden. Sei $K = \max(M, N)$.

Durch Erweitern der Gleichung (3.4.4) mit z^K folgt:

$$G(z) = \frac{z^{N-M} * \sum_{k=0}^M b_{M-k} * z^k}{\sum_{k=0}^N a_{N-k} * z^k} = \frac{Z(z)}{N(z)} \quad (3.4.10)$$

Wegen $a_0 = 1$ hat das Nennerpolynom $N(z)$ den Grad K . Der Grad des Zählerpolynoms $Z(z)$ ist $\leq K$. Ein realisierbares Filter besitzt also eine Übertragungsfunktion, in der der Zählergrad kleiner oder gleich dem Nennergrad ist.

Eine Übertragungsfunktion, deren Zählergrad größer als der Nennergrad ist, führt zu einem Filter, das zukünftige Werte benutzt und deshalb nicht realisierbar ist.

Eine Übertragungsfunktion, deren Zählergrad kleiner als der Nennergrad ist, führt zu einem Filter, das zumindest den gegenwärtigen Eingangswert x_n nicht benutzt und deshalb ein Totzeitglied enthält.

Bei digitalen Regelungen treten häufig Reglertotzeiten auf, die gleich der Abtastzeit sind.

Bei der Behandlung dieser Reglertotzeit sind zwei Wege gangbar:

a) Die Reglertotzeit wird der Strecke zugerechnet. Dies führt zu einem zusätzlichen Faktor $1/z$ in der Übertragungsfunktion der Strecke.

b) Die Reglertotzeit wird im Reglergesetz dadurch berücksichtigt, daß der gegenwärtige Eingangswert x_n nicht benutzt wird.

Im Reglergesetz (3.4.1) muß dann $b_0 = 0$ sein, woraus folgt, daß Zählergrad der Regler-ÜF kleiner als der Nennergrad ist.

3.5 Z-ÜBERTRAGUNGSFUNKTIONEN EINFACHER DIGITALER REGLER

Im folgenden sollen für einige einfache digitale Regler die Reglergesetze und die zugehörigen Z-Übertragungsfunktionen angegeben werden. Digitale zeitdiskrete Regler oder Filter verhalten sich nur bei Eingangsfrequenzen, die wesentlich kleiner als die Abtastfrequenz ist, so wie die entsprechenden analogen zeitkontinuierlichen Filter.

1. Integrator Rechteckregel

$$y_n = y_{n-1} + kT \cdot x_{n-1} \quad (3.5.1)$$

$$G(z) = \frac{kT}{z-1}$$

2. Integrator Trapezregel

$$y_n = y_{n-1} + kT/2 \cdot (x_n + x_{n-1}) \quad (3.5.2)$$

$$G(z) = \frac{kT}{2} \cdot \frac{z+1}{z-1}$$

3. Differenzierer Sekantenregel

$$y_n = (x_n - x_{n-1})/kT \quad (3.5.3)$$

$$G(z) = \frac{z-1}{kTz}$$

4. Differenzierer mit quadratischer Interpolation

$$y_n = (3x_n - 2x_{n-1} + x_{n-2})/2kT \quad (3.5.4)$$

$$G(z) = \frac{3z^2 - 2z + 1}{kTz^2}$$

Zur Ableitung der angegebenen Formel wird der folgende Ansatz gemacht:

$$x(\tau) = a \cdot \tau^2 + b \cdot \tau + c, \quad x_n = x(2), \quad x_{n-1} = x(1), \quad x_{n-2} = x(0)$$

Hieraus folgt:

$$c = x_{n-2}, \quad x_{n-1} = a + b + c, \quad x_n = 4a + 2b + c$$

$$a = x_n/2 - x_{n-1} + x_{n-2}/2$$

$$b = 2x_{n-1} - x_n/2 - 3/2 x_{n-2}$$

Die Differentiation an der Stelle $\tau=2$ liefert:

$$\left. \frac{dx}{d\tau} \right|_{\tau=2} = 2a\tau + b = 4a + b = 1/2 * (3x_n - 2x_{n-1} + x_{n-2})$$

Unter Berücksichtigung der Substitution $\tau = t/T$ folgt das obige Resultat.

3.6 Z-ÜBERTRAGUNGSFUNKTION UND SYSTEMEIGENSCHAFTEN

Aus der Z-Übertragungsfunktion lassen sich folgende Schlüsse hinsichtlich des Systemverhaltens ziehen:

1. Stabilität

Das System ist genau dann BIBO-stabil, wenn die Pole der Z-ÜF innerhalb des Einheitskreises liegen /3 HARTMANN/.

Die Methode der Polvorgabe beim Reglerentwurf stellt also sicher, daß die Reglerkoeffizienten so gewählt sind, daß das System zumindest BIBO-stabil ist.

2. stationäres Endwertverhalten

Sei $G_F(z) = Z_F(z)/N_F(z)$ die Führ-ÜF des geschlossenen Regelkreises.

Durch Anwendung des Endwertsatzes der Z-Transformation kann das Endwertverhalten des Systems bestimmt werden /3 HARTMANN S.5.8/.

Es ist dabei günstig, die Endwertfehler-Funktion

$$G_E(z) = 1 - G_F(z) = (N_F(z) - Z_F(z))/N_F(z) \quad (3.6.1)$$

zu betrachten.

Im folgenden wird vorausgesetzt, daß $N_F(1) = 0$ ist. Dies bedeutet, daß keine Polstelle von $G_F(z)$ auf dem Rand des Einheitskreises liegt.

Der bei der Systemantwort auf eine Sprungfunktion auftretende bleibende Regelfehler wird als Lagefehler e_1 bezeichnet.

Der Fehler e_1 wird genau dann Null, wenn $Z_F(z)$ durch $(z-1)$ teilbar ist. Dies ist gleichbedeutend mit $Z_E(1) = 0$ und $G_F(1) = 1$.

Der bei der Systemantwort auf eine Rampenfunktion $r(t) = t$ auf-

tretende bleibende Regelfehler wird als Geschwindigkeitsfehler e_2 bezeichnet.

Der Fehler e_2 wird genau dann Null, wenn $Z_E(z)$ durch $(z-1)^2$ teilbar ist.

Der bei der Systemantwort auf die Führungsfunktion $a(t) = t^2$ auftretende bleibende Regelfehler wird als Beschleunigungsfehler e_3 bezeichnet.

Der Fehler e_3 wird genau dann Null, wenn $Z_E(z)$ durch $(z-1)^3$ teilbar ist.

Für die Stör-ÜF $G_D(z)$ eines geschlossenen Regelkreises gilt mit der auf einen Einfachregelkreis umgerechneten ÜF $G_C(z)$ des Reglers:

$$G_D(z) = G_F(z) / G_C(z) \quad (3.6.2)$$

Damit der bei einem Störgrößensprung auftretende bleibende Regelfehler Null wird, muß gelten:

$$G_D(1) = 1 \quad (3.6.3)$$

Mit $G_F(1) = 1$ folgt hieraus die Bedingung $G_C(1) = 1$.

3. Zeitverhalten - Sprungantwort

Das Zeitverhalten des Systems läßt sich aus der Z-ÜF nicht unmittelbar ablesen.

Das Auftreten von komplexen Polstellenpaaren führt zum Auftreten von gedämpften Sinustermen im Zeitverhalten. Dabei kann jedoch die Amplitude so klein sein oder die Dämpfung so groß sein, daß diese Sinusterme nicht mehr in Erscheinung treten.

Um die Antwort des Systems auf eine Führungsfunktion $r(t)$ zu erhalten, sind folgende Schritte notwendig:

- a) Bildung der Funktion $y(z) = G(z) * r(z)$
- b) Faktorisierung der Funktion $y(z)$
- c) Partialbruchzerlegung der Funktion $y(z)$
- d) Durchführung der inversen Z-Transformation,
d.h. Bildung der Funktion $y(t)$
- e) Beurteilung des Zeitverlaufs $y(t)$, ($0 \leq t \leq T_{max}$) nach vorge-

gegebenen Kriterien.

Die manuelle Durchführung dieser Schritte ist sehr zeitaufwendig. Bei der Arbeit mit Rechnerunterstützung und Verfügbarkeit eines entsprechenden Programms können diese Schritte vom Rechner durchgeführt werden.

Einfacher und numerisch genauer ist jedoch bei der Arbeit mit Rechnerunterstützung die Verwendung des Modells im Zeitbereich zur Gewinnung der Funktion $y(t)$. Diese Simulation, die der Rechner in der kurzen Zeit durchführen kann, gestattet zudem noch die Berücksichtigung von nichtlinearen Elementen des realen Systems, wie z.B. Begrenzungen und begrenzte Genauigkeit bei Operationen, die vom digitalen Regler durchgeführt werden. Die Beurteilung des Zeitverlaufs nach vorgegebenen Kriterien kann ebenfalls vom Rechner durchgeführt werden, so daß ohne Eingreifen des Benutzers eine große Zahl von Koeffizientenkombinationen in ihrer Auswirkung auf das Zeitverhalten des Systems untersucht werden kann.

Die Sprungantwort eines Systems kann nach folgenden Kriterien beurteilt werden:

- Zeitdauer bis zum Erreichen des Sollwerts bei Vorgabe einer zulässigen Sollwertabweichung
- Überschwingweite
- Zeitdauer bis zum Ausregeln eines Störgrößensprungs bei Vorgabe einer zulässigen Sollwertabweichung
- Stellgrößenaufwand.

4. Frequenzverhalten - Frequenzkennlinien

Aus der Z-ÜF $G(z)$ entsteht bekanntlich durch Einsetzen von:

$$z = e^{sT} \quad (3.6.4)$$

die zugehörige S-ÜF $G^*(s) = G(e^{sT})$. Die Betrags- und die Phasenkennlinien entstehen durch Einsetzen von $s = j\omega$.

Beim Arbeiten ohne Rechnerunterstützung ist die Tatsache hinderlich, daß die Funktion $G(s)$ keine rationale Funktion ist. Deshalb verwendet man hier anstelle der Transformation $z = e^{sT}$ die sogenannte bilineare Transformation:

$$z = B(p) = \frac{1 + p}{1 - p} \quad (3.6.5)$$

Um den Unterschied zwischen den Transformationen zu betonen, wird hier anstelle der Variablen s die Variable p benutzt.

Die bilineare Transformation, die einen Spezialfall der in der Mathematik auch unter dem Namen Kugeldrehungen bekannten Transformationen darstellt, besitzt für den Bereich $p = j\omega$ ähnliche Eigenschaften wie die Transformation $z = e^{sT}$.

Für $z = e^{sT}$ gilt mit $s = j\omega$:

$$\begin{aligned} |z| &= 1 \\ \arg(z) &= \omega T \end{aligned} \quad (3.6.6)$$

Für $z = B(p)$ gilt mit $p = j\Omega$:

$$\begin{aligned} |z| &= 1 \\ \arg(z) &= \arctan(2\Omega/(1-\Omega^2)) = 2 \arctan(\Omega) \end{aligned} \quad (3.6.7)$$

Gleichsetzen von (3.6.6) und (3.6.7) liefert:

$$2 \arctan(\Omega) = \omega T$$

Hieraus folgt:

$$\Omega = \tan(\omega T/2) \quad (3.6.8)$$

Interessant ist hier aufgrund des Abtasttheorems nur der Bereich:

$$0 \leq \omega \leq 2\pi/(2T) \quad (3.6.9)$$

Nach Bildung der rationalen Funktion $G^*(j\Omega) = G(B(j\Omega))$ können somit unter Berücksichtigung von (3.6.8) die Frequenzkennlinien gezeichnet werden.

Beim Arbeiten mit Rechnerunterstützung ist die Verwendung der bilinearen Transformation entbehrlich. Die Berechnung der Funktion $G^*(j\omega)$ an entsprechend vielen Stellen ermöglicht hier unmittelbar eine graphische Darstellung der Frequenzkennlinien.

Auch die Bewertung der Frequenzkennlinien kann dem Rechner übertragen werden.

Bei der Führungs-ÜF $G_F^*(j\omega)$ ist erwünscht:

$$|G_F^*(j\omega)| \approx 1 \quad (3.6.10)$$

für einen möglichst großen Frequenzbereich.

Bei der Stör-ÜF $G_T^*(j\omega)$ ist erwünscht:

$$|G_T^*(j\omega)| \ll 1 \quad (3.6.11)$$

für einen möglichst großen Frequenzbereich.

3.7 REGLERENTWURF MIT DER Z-ÜBERTRAGUNGSFUNKTION

3.7.1 DIE Z-ÜBERTRAGUNGSFUNKTION DER STRECKE

Betrachtet werden hier zeitdiskrete lineare Systeme mit einer Einganggröße u und einer zu regelnden Ausgangsgröße y .

Es wird folgendes Zustandsmodell zugrundegelegt:

$$\begin{aligned} x_{i+1} &= \Phi * x_i + H * u_i & x_i &\in \mathbb{R}^n \\ y_i &= C * x_i \end{aligned} \quad (3.7.1.1)$$

Durch Z-Transformation folgt:

$$\begin{aligned} z * (X(z) - x_0) &= \Phi * X(z) + H * U(z) \\ Y(z) &= C * X(z) \end{aligned}$$

Durch Umformung erhält man die Ausgangsbeziehung:

$$Y(z) = z * C * (E * z - \Phi)^{-1} * x_0 + C * (E * z - \Phi)^{-1} * H * U(z)$$

Mit $x_0 = 0$ folgt für die ÜF $G_S(z)$ der Strecke:

$$G_S(z) = C * (E * z - \Phi)^{-1} * H \quad (3.7.1.2)$$

Hieraus erhält man:

$$G_S(z) = \frac{C * \text{adj}(E * z - \Phi) * H}{\det(E * z - \Phi)} = \frac{Z_S(z)}{N_S(z)} \quad (3.7.1.3)$$

Dabei ist $N_s(z)$ ein Polynom vom Grade n und $Z_s(z)$ ein Polynom, dessen Grad $< n$ ist.

3.7.2 DIE FÜHRUNGS-ÜBERTRAGUNGSFUNKTION

Werden bei einem System die Reglerübertragungsfunktionen mit unbekannten Koeffizienten angesetzt, so gelingt es mit einigem Aufwand, die Führungs-ÜF $G_F(z)$ und die Stör-ÜF $G_D(z)$ in Abhängigkeit von den Koeffizienten zu berechnen ($f_N=1$) :

$$G_F(z) = \frac{\sum_{i=0}^M h_i * z^i}{\sum_{i=0}^N f_i * z^i} = \frac{Z_F(z)}{N_F(z)} \quad (3.7.2.1)$$

Darin sind die Koeffizienten h_i und f_i Funktionen der unbekannten Reglerkoeffizienten. Werden die Reglerkoeffizienten in dem Koeffizientenvektor k zusammengefaßt, so gilt also:

$$h_i = h_i(k) \quad (i=0, \dots, M)$$

und

$$f_i = f_i(k) \quad (i=0, \dots, N-1)$$

Gibt man die Pole $p_1 \dots p_N$ vor, so gilt:

$$N(z) = (z - p_1) * (z - p_2) * \dots * (z - p_N) \quad (3.7.2.2)$$

Durch Ausmultiplizieren entsteht der Ausdruck:

$$N(z) = g_0 + g_1 * z + \dots + z^N \quad (3.7.2.3)$$

Die unbekannten Koeffizienten k_i können also durch Lösung der $N-1$ Gleichungen

$$g_i = f_i(k) \quad (i=0, \dots, N-1)$$

gefunden werden.

3.7.3 ENTWURF EINES EINFACHREGELKREISES

Betrachtet wird eine Reglerstruktur nach Bild 3.1 .

Ist $G_c(z)$ die ÜF des Reglers, so gilt:

$$G_F(z) = \frac{G_c(z) * G_s(z)}{1 + G_c(z) * G_s(z)} = \frac{Z_c(z) * Z_s(z)}{Z_c(z) * Z_s(z) + N_c(z) * N_s(z)} = \frac{Z_F(z)}{N_F(z)} \quad (3.7.3.1)$$

und

$$G_B(z) = \frac{G_s(z)}{1 + G_c(z) * G_s(z)} = \frac{N_c(z) * Z_s(z)}{Z_c(z) * Z_s(z) + N_c(z) * N_s(z)} = \frac{Z_B(z)}{N_B(z)} \quad (3.7.3.2)$$

Gibt man das Polynom $N_F(z)$ vor, so können die Polynome $N_c(z)$ und $Z_c(z)$ aus der Gleichung

$$Z_c(z) * Z_s(z) + N_c(z) * N_s(z) = N_F(z) \quad (3.7.3.3)$$

bestimmt werden.

Sei $\text{grad}(Z_s(z)) < \text{grad}(N_s(z)) = N$ und $\text{grad}(N_c(z)) = \text{grad}(Z_c(z)) = K$.

Mit $\text{koeff}(N_s(z), N) = \text{koeff}(N_c(z), K) = \text{koeff}(N_F(z), N+K) = 1$ folgt, daß für die Berechnung der $2K+1$ Reglerkoeffizienten $N+K$ Gleichungen zur Verfügung stehen. Das Gleichungssystem ist also überbestimmt für $K < N-1$, bestimmt für $K = N-1$ und unterbestimmt für $K > N-1$.

Die Wahl von $K > N-1$ ist sinnvoll, da in diesem Falle die gezielte Beeinflussung von $Z_F(z)$ und $Z_B(z)$ zur Erreichung eines gewünschten Endwertverhaltens möglich ist.

Die Ordnung des geschlossenen Regelkreises ist:

$$\text{grad}(N_F(z)) = N+K \quad (3.7.3.4)$$

Es ist $\text{grad}(Z_F(z)) = \text{grad}(Z_B(z)) = \text{grad}(Z_s(z)) + K < N+K$

Liegen Nullstellen von $Z_s(z)$ im Einheitskreis, so ist es möglich und anzustreben, diese durch Nullstellen von $N_F(z)$ zu kompensieren.

sieren, da sich hierdurch die Ordnung des Regelkreises verringert
/3 HARTMANN S.5.14/, /10 SCHÖNFELD/.

3.7.4 ENTWURF EINES P-ZUSTANDSREGLERS

Das Reglergesetz lautet hier /3 HARTMANN S.5.30/:

$$u_1 = K(F \cdot r_1 - x_1) = \sigma \cdot r_1 - K \cdot x_1 \quad (3.7.4.1)$$

Die Struktur dieses Reglers ist in Bild 3.2 gezeigt.

Mit dem Zustandsmodell (3.7.1.1) der Strecke folgt hieraus:

$$x_{1+1} = (\Phi - H \cdot K) \cdot x_1 + H \cdot \sigma \cdot r_1 \quad (3.7.4.2)$$

Z-Transformation ergibt mit $x_0 = 0$:

$$(z \cdot E - \Phi + H \cdot K) \cdot X(z) = H \cdot \sigma \cdot R(z)$$

Mit der Ausgangsgleichung $Y(z) = C \cdot X(z)$ folgt:

$$Y(z) = C \cdot (z \cdot E - \Phi + H \cdot K)^{-1} \cdot H \cdot \sigma \cdot R(z)$$

Die Führungs-ÜF lautet also:

$$G_F(z) = C \cdot (z \cdot E - \Phi + H \cdot K)^{-1} \cdot H \cdot \sigma \quad (3.7.4.3)$$

$$G_F(z) = \frac{\sigma \cdot C \cdot \text{adj}(E \cdot z - \Phi + H \cdot K) \cdot H}{\det(E \cdot z - \Phi + H \cdot K)} = \frac{Z_F(z)}{N_F(z)} \quad (3.7.4.4)$$

Bei einem Zustandsmodell der Ordnung N ist also $\text{grad}(N_F(z)) = N$
und $\text{grad}(Z_F(z)) = N-1$.

Die Berücksichtigung von Störeinflüssen im Zustandsmodell erfolgt
durch den Ansatz:

$$x_{1+1} = \Phi \cdot x_1 + H \cdot u_1 + D \cdot d_1 \quad (3.7.4.5)$$

Dabei ist D ein Störgrößeneinflußvektor.

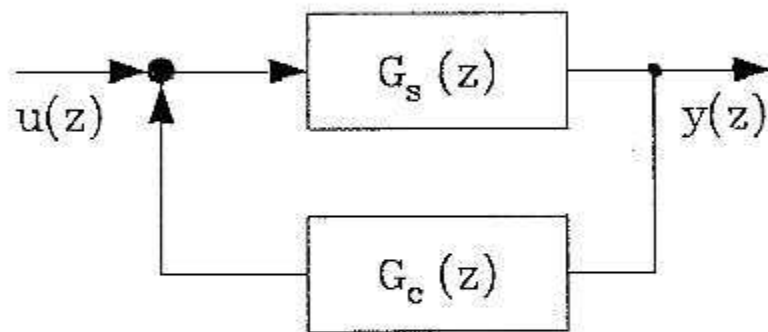


Bild 3.1 Strukturbild eines Einfachregelkreises

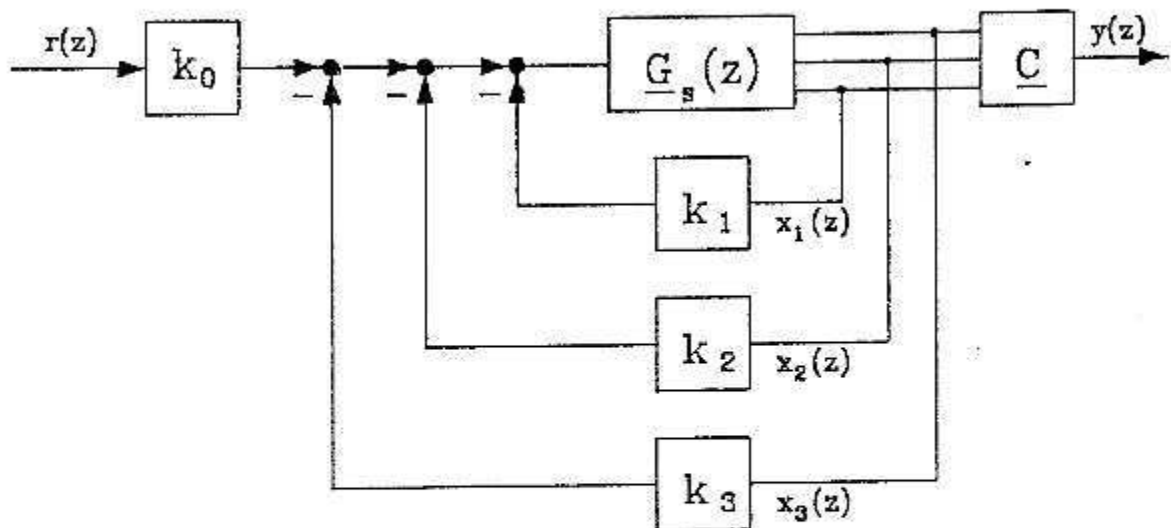


Bild 3.2 Strukturbild eines Regelkreises mit P-Zustandsregler

Für die Stör-ÜF folgt damit:

$$G_D(z) = C \cdot (z \cdot E - \Phi + H \cdot K)^{-1} \cdot D = \frac{C \cdot \text{adj}(E \cdot z - \Phi + H \cdot K) \cdot D}{N_F(z)} \quad (3.7.4.6)$$

Bei der Vorgabe von $N_F(z)$ ist der Vektor K so zu bestimmen, daß gilt:

$$\det(E \cdot z - \Phi + H \cdot K) = N_F(z) \quad (3.7.4.7)$$

Für eine beliebige Matrix A gilt /3 HARTMANN S. 5.32/ :

$$\det(A + H \cdot K) = \det(A) + K \cdot \text{adj}(A) \cdot H \quad (3.7.4.8)$$

Damit folgt:

$$K \cdot \text{adj}(E \cdot z - \Phi) \cdot H = N_F(z) - N_S(z) \quad (3.7.4.9)$$

Aus dieser Gleichung kann K durch Koeffizientenvergleich bestimmt werden.

Weiterhin gilt für eine beliebige Matrix A /3 HARTMANN S.5.42 /:

$$\text{adj}(A + H \cdot K) \cdot H = \text{adj}(A) \cdot H \quad (3.7.4.10)$$

Damit erhält man:

$$C \cdot \text{adj}(E \cdot z - \Phi + H \cdot K) \cdot H = C \cdot \text{adj}(E \cdot z - \Phi) \cdot H = Z_S(z) \quad (3.7.4.11)$$

Daraus folgt:

$$Z_F(z) = \sigma \cdot Z_S(z) \quad (3.7.4.12)$$

und:

$$G_F(z) = \frac{Z_F(z)}{N_F(z)} = \frac{\sigma \cdot Z_S(z)}{K \cdot \text{adj}(E \cdot z - \Phi) \cdot H + N_S(z)} \quad (3.7.4.13)$$

Wird nun $\sigma = N_F(1)/Z_S(1)$ gewählt, so gilt $G_F(1) = 1$ und es folgt, daß $e_1 = 0$ ist.

3.7.5 ENTWURF EINES PI-ZUSTANDSREGLERS

Man strebt hier an, ein erweitertes Zustandsmodell zu finden. Das erweiterte Zustandsmodell soll dabei so beschaffen sein, daß beim Entwurf eines P-Reglers für dieses Zustandsmodell die Konstanten des PI-Zustandsreglers entstehen.

Die Erweiterung kann am einfachsten dadurch erfolgen, daß das Integral w_i als neue Zustandsgröße der Strecke aufgefaßt wird. Die Formel für die Berechnung des Integrals w_i lautet:

$$w_i = w_{i-1} + (r_i - y_i) \quad (3.7.5.1)$$

Diese Formel kann jedoch nicht verwendet werden, wenn das Integral w_i unmittelbar als Zustandsgröße der Strecke zugeordnet wird.

Anstelle der Gleichung (3.7.5.1) muß dann die Gleichung:

$$w_i = w_{i-1} + (r_{i-1} - y_{i-1}) \quad (3.7.5.2)$$

verwendet werden.

Die Verwendung der Formel (3.7.5.1) ist jedoch möglich, wenn zunächst die Führungs-ÜF des PI-Regelkreises berechnet wird, und anschließend durch mathematische Umformungen ein erweitertes Zustandsmodell bestimmt wird. Beide Wege sollen hier beschriftet werden.

Bild 3.3 zeigt die Struktur des PI-Zustandsreglers.

Die Größen im erweiterten Modell werden durch den tiefgestellten Index i gekennzeichnet.

a) Bestimmung eines erweiterten Zustandsmodells mit der Z-ÜF
Das Reglergesetz lautet /3 HARTMANN S.5.30/:

$$u_i = -K \cdot x_i + k_1 \cdot w_i + k_0 \cdot r_i \quad (3.7.5.3)$$

Z-Transformation ergibt:

$$W = (1 - z^{-1})^{-1} \cdot (R - Y)$$

$$U = -K*X + k_I * W + k_0 * R$$

Mit

$$G_I(z) = z/(z-1)$$

erhält man:

$$U = -K*X + k_I * G_I * (R - Y) + k_0 * R$$

$$U = -K*X - k_I * G_I * Y + (k_I * G_I + k_0) * R$$

Mit dem transformierten Zustandsmodell der Strecke folgt:

$$(z*E - \Phi) * X = H * (-K*X - k_I * G_I * Y + (k_I * G_I + k_0) * R)$$

$$(z*E - \Phi + H*K) * X = H * (-k_I * G_I * Y + (k_I * G_I + k_0) * R)$$

Mit der Ausgangsgleichung $Y(z) = C*X(z)$ folgt:

$$Y = C * (z*E - \Phi + H*K)^{-1} * H * (-k_I * G_I * Y + (k_I * G_I + k_0) * R)$$

Mit der Abkürzung

$$A = z*E - \Phi + H*K \quad (3.7.5.4)$$

folgt:

$$Y * (1 + C*A^{-1} * H * k_I * G_I) = C*A^{-1} * H * (k_I * G_I + k_0) * R$$

$$Y * (z-1 + C*A^{-1} * H * k_I * z) = C*A^{-1} * H * (k_I * z + k_0 * (z-1)) * R$$

$$Y * ((z-1) * \det(A) + C * \text{adj}(A) * H * k_I * z) =$$

$$= C * \text{adj}(A) * H * (k_I * z + k_0 * (z-1)) * R$$

$$G_{1f}(z) = \frac{C * \text{adj}(A) * H * (k_I * z + k_0 * (z-1))}{(z-1) * \det(A) + C * \text{adj}(A) * H * k_I * z} \quad (3.7.5.5)$$

Mit der Matrix B:

$$B = A + H * C \quad (3.7.5.6)$$

erhält man unter Verwendung der Gleichungen (3.7.4.8) und

(3.7.4.10) :

$$\begin{aligned}
 N_{IF} &= (z-1) * (\det(A) + C * \text{adj}(A) * H * k_I) + C * \text{adj}(A) * H * k_I \\
 &= (z-1) * \det(B) + C * \text{adj}(B) * H * k_I \\
 &= (z-1) * \det(B) + C * B^{-1} * \det(B) * H * k_I \\
 &= \det(B) * ((z-1) + C * B^{-1} * H * k_I) = \det(B_I)
 \end{aligned}$$

Dabei ist die Matrix B_I definiert durch:

$$B_I = \begin{bmatrix} B & -k_I * H \\ C & z-1 \end{bmatrix} = \begin{bmatrix} A+H*C & -k_I * H \\ C & z-1 \end{bmatrix} = \begin{bmatrix} z * E - \Phi + H * (K+C) & -k_I * H \\ C & z-1 \end{bmatrix}$$

Mit:

$$\Phi_I = \begin{bmatrix} \Phi - H * C & 0 \\ -C & 1 \end{bmatrix} \quad H_I = \begin{bmatrix} H \\ 0 \end{bmatrix} \quad K_I = [K, k_I]$$

folgt:

$$B_I = z * E_I - \Phi_I + H_I * K_I$$

Für $Z_{IF}(z)$ erhält man mit $C_I = [C, 0]$:

$$\begin{aligned}
 Z_{IF}(z) &= C * \text{adj}(A) * H * (k_I * z + k_0 * (z-1)) \\
 &= C * \text{adj}(B) * H * (k_I * z + k_0 * (z-1)) \\
 &= C_I * \text{adj}(B_I) * H_I * (k_I * z + k_0 * (z-1))
 \end{aligned}$$

Mit:

$$C * \text{adj}(A) * H = Z_S(z) = Z_{IS}(z)$$

folgt:

$$Z_{IF}(z) = Z_{IS}(z) * ((k_I + k_0) * z - k_0)$$

Damit ist über die Z-ÜF ein erweitertes Zustandsmodell gefunden, das den Reglerentwurf nach der Methode des P-Zustandsreglerentwurfs ermöglicht.

Unter Benutzung von (3.7.1.3) erhält man:

$$G_{IF}(z) = \frac{Z_S(z) * (k_I * z + k_0 * (z-1))}{(z-1) * N_F(z) + Z_S(z) * k_I * z} = \frac{Z_{IF}(z)}{N_{IF}(z)} \quad (3.7.5.7)$$

$$G_{IF}(z) = \frac{Z_S * (k_I * z + k_0 * (z-1))}{(z-1) * (K * \text{adj}(E * z - \Phi) * H + N_S) + Z_S * k_I * z} \quad (3.7.5.8)$$

Wird nun das Polynom $N_F(z)$ vorgegeben, so kann K nach (3.7.4.9) bestimmt werden. Mit k_I kann das Polynom $N_{IF}(z)$ beeinflusst werden. Ausgehend von $Z_S(z)$ kann mit k_I und k_0 das Polynom $Z_{IF}(z)$ beeinflusst werden.

Für $k_I = 0$ gilt mit $k_0 = \sigma$: $G_{IF}(z) = G_F(z)$

Für $Z_E = N_{IF} - Z_{IF}$ gilt:

$$Z_E = (z-1) * (N_F - Z_F * k_0) \quad (3.7.5.9)$$

Unter der Bedingung $N_{IF}(1) \neq 0$ gilt also $Z_E(1) = 0$ und damit $e_1 = 0$. Mit $k_0 = N_F(1)/Z_F(1)$ kann erreicht werden, daß auch $e_2 = 0$ wird.

b) direkte Erweiterung des Zustandsmodells

Zugrundegelegt wird hier die Formel (3.7.5.2).

Das Zustandsmodell wird nun um die Zustandsgröße w_1 erweitert.

Hierzu sind folgende Schritte notwendig:

Zum Zustandvektor wird das Element w_1 hinzugefügt.

Zur Transitionsmatrix wird die Zeile $[-C, 1]$ und die Spalte $[0^T, 1]^T$ hinzugefügt.

Die Vektoren C und H werden je durch Hinzufügen einer Null erweitert.

Die Größen im erweiterten Modell werden wieder durch den tiefgestellten Index I gekennzeichnet.

Der Vektor H_R wird definiert durch $H_R = [0^T, 1]^T$.

$$\Phi_I = \begin{bmatrix} \Phi & 0 \\ -C & 1 \end{bmatrix} \quad H_I = \begin{bmatrix} H \\ 0 \end{bmatrix} \quad K_I = [K, k_I]$$

Das transformierte Zustandsmodell lautet dann:

$$z^*X_I = \Phi_I * X_I + H_I * U + H_R * R$$

Für das erweiterte Modell gilt mit $R=0$:

$$N_{Is}(z) = \det(z^*E_I - \Phi_I) = (z-1) * \det(z^*E - \Phi) = (z-1) * N_s(z)$$

$$Z_{Is}(z) = C_I * \text{adj}(z^*E_I - \Phi_I) * H_I$$

Das erweiterte Zustandsmodell ist also von der Ordnung $N+1$.

Mit dem transformierten Reglergesetz:

$$U = -K_I * X_I + k_0 * R$$

folgt:

$$z^*X_I = \Phi_I * X_I + H_I * (-K_I * X_I + k_0 * R) + H_R * R$$

$$z^*X_I = (\Phi_I - H_I * K_I) * X_I + (H_I * k_0 + H_R) * R$$

$$(z^*E_I - \Phi_I + H_I * K_I) * X_I = (H_I * k_0 + H_R) * R$$

$$Y = C_I * (z^*E_I - \Phi_I + H_I * K_I)^{-1} * (H_I * k_0 + H_R) * R$$

Mit der Abkürzung

$$A_I = z^*E_I - \Phi_I + H_I * K_I$$

folgt:

$$G_{IF}(z) = \frac{C_I * \text{adj}(A_I) * (H_I * k_0 + H_R)}{\det(A_I)} \quad (3.7.5.10)$$

Ausgehend von diesem Zustandsmodell kann nun ein P-Zustandsregler entworfen werden.

Die Bestimmung des Vektors K_I erfolgt dabei nach der Gleichung:

$$K_I * \text{adj}(E_I * z - \Phi_I) * H_I = N_{IF}(z) - N_{Is}(z) \quad (3.7.5.11)$$

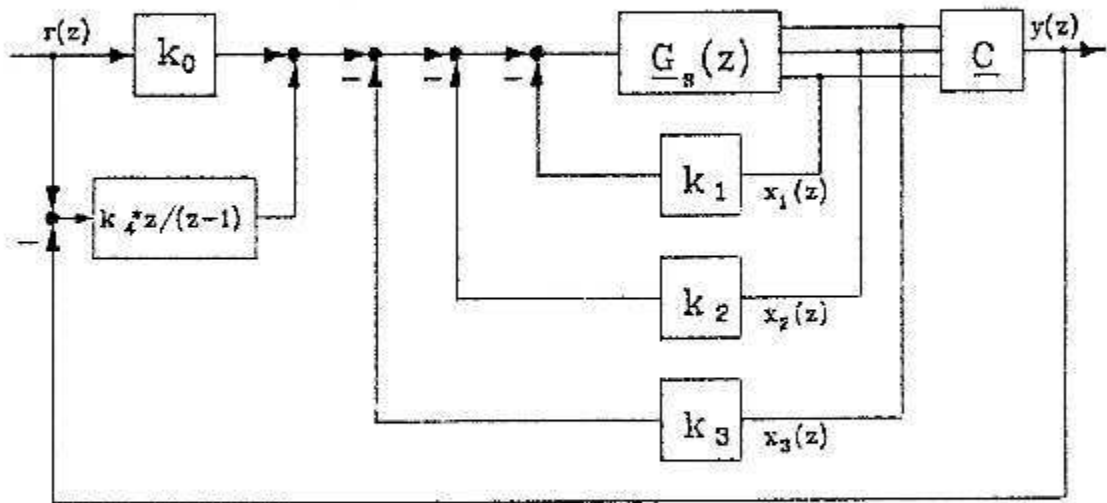


Bild 3.3 Strukturbild eines Regelkreises mit PI-Zustandsregler

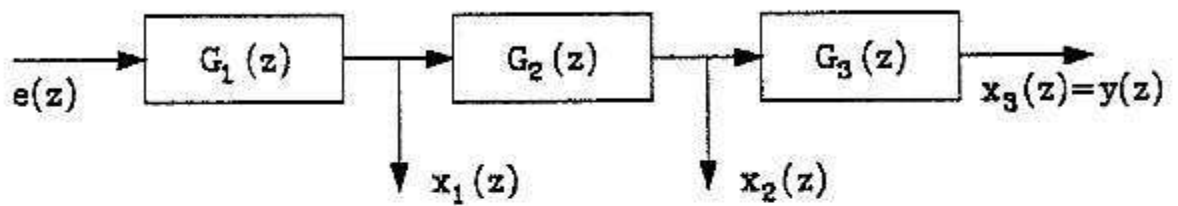


Bild 3.4 Strukturbild einer Strecke mit Blockstruktur

Für $Z_{IF}(z)$ folgt:

$$Z_{IF}(z) = C_I \cdot \text{adj}(A_I) \cdot H_I \cdot k_0 + C_I \cdot \text{adj}(A_I) \cdot H_R$$

$$Z_{IF}(z) = Z_S(z) \cdot k_0 + C_I \cdot \text{spalte}(\text{adj}(A_I), N+1) \quad (3.7.5.12)$$

3.7.6 ENTWURF EINES ZUSTANDSREGLERS MIT TOTZEITBERÜCKSICHTIGUNG

Es wird hier angenommen, daß die durch die Rechenzeit bedingte Reglertotzeit gleich der Abtastzeit T ist. Das Totzeitglied wird an den Streckeneingang gelegt. Die um T verzögerte Stellgröße $u_{T1} = u_{1-1}$ stellt eine neue Zustandsgröße dar.

Das Zustandsmodell wird erweitert, indem zum Zustandvektor das Element u_{T1} hinzugefügt wird und zur Transitionsmatrix die Zeile $[0]$ und die Spalte $[H', 0]'$ hinzugefügt wird.

Die Größen im erweiterten Modell werden durch den tiefgestellten Index T gekennzeichnet.

Dann gilt:

$$C_T = [C_I, 0] \quad \text{und} \quad H_T = [0^T, 1]^T$$

Für das erweiterte Modell gilt:

$$G_{TS}(z) = Z_{TS}(z) / N_{TS}(z)$$

mit

$$N_{TS}(z) = \det(z \cdot E_T - \Phi_T) = z \cdot \det(z \cdot E - \Phi) = z \cdot N_S(z) \quad (3.7.6.1)$$

und

$$Z_{TS}(z) = C_T \cdot \text{spalte}(\text{adj}(z \cdot E_T - \Phi_T), N+1) \quad (3.7.6.2)$$

Für dies erweiterte Zustandsmodell kann ein P- oder PI-Zustandsregler entworfen werden.

3.7.7 ZUSTANDSREGLERENTWURF BEI EINER STRECKE MIT BLOCKSTRUKTUR

Betrachtet wird im folgenden eine Strecke mit drei Zustandsgrößen. Dies stellt jedoch keine Einschränkung hinsichtlich der Allgemeinheit der Ausführungen dar. Die betrachtete Strecke besitze eine Blockstruktur gemäß Bild 3.4. Dann vereinfacht sich der Zustandsreglerentwurf beträchtlich.

Es gilt:

$$G_S(z) = G_1(z) * G_2(z) * G_3(z) \quad (3.7.7.1)$$

Bezeichnet $E(z)$ die Eingangsfunktion der Strecke und $Y(z)$ die Ausgangsfunktion der Strecke, so gilt:

$$X_1(z) = G_1(z) * E(z) = G_3(z)^{-1} * G_2(z)^{-1} * Y(z) \quad (3.7.7.2a)$$

$$X_2(z) = G_1(z) * G_2(z) * E(z) = G_3(z)^{-1} * Y(z) \quad (3.7.7.2b)$$

$$X_3(z) = G_S(z) * E(z) = Y(z) \quad (3.7.7.2c)$$

Ein P-Zustandsregler besitzt das transformierte Reglergesetz:

$$E(z) = k_0 * R(z) - k_1 * X_1(z) - k_2 * X_2(z) - k_3 * X_3(z)$$

Unter Benutzung von (3.7.7.2) folgt:

$$G_S^{-1} * Y = k_0 * R - (k_1 * G_3^{-1} * G_2^{-1} + k_2 * G_3^{-1} + k_3) * Y$$

$$(G_S^{-1} + k_1 * G_3^{-1} * G_2^{-1} + k_2 * G_3^{-1} + k_3) * Y = k_0 * R$$

$$(1 + k_1 * G_1 + k_2 * G_1 * G_2 + k_3 * G_S) * Y = k_0 * G_S * R$$

$$G_F = \frac{k_0 * G_S}{1 + k_1 * G_1 + k_2 * G_1 * G_2 + k_3 * G_S}$$

$$G_F = \frac{k_0 * Z_S}{N_S + k_1 * Z_1 * N_2 * N_3 + k_2 * Z_1 * Z_2 * N_3 + k_3 * Z_S} \quad (3.7.7.3)$$

Der Reglerentwurf erfolgt ausgehend von der Bedingung:

$$N_F = N_S + k_1 * Z_1 * N_2 * N_3 + k_2 * Z_1 * Z_2 * N_3 + k_3 * Z_S \quad (3.7.7.4)$$

Da $\text{grad}(N_F) = \text{grad}(N_S) = 3$ ist, sind die Konstanten k_1, k_2 und k_3 durch (3.7.7.4) eindeutig bestimmt.

Das Zählerpolynom $Z_F = k_0 * Z_S$ kann durch die Polstellenvorgabe nicht beeinflußt werden.

Die Konstante k_0 wird durch die Endwertbedingung $G_F(1) = 1$ festgelegt:

$$k_0 = N_F(1) / Z_S(1) \quad (3.7.7.5)$$

Die Berücksichtigung der Totzeit erfolgt, indem anstelle von $G_1(z)$ die Funktion $G_{T1}(z) = z^{-1} * G_1(z)$ verwendet wird.

Ein PI-Zustandsregler besitzt das transformierte Reglergesetz:

$$E = k_0 * R - k_1 * X_1 - k_2 * X_2 - k_3 * X_3 + k_1 * z * (z-1)^{-1} * (R - Y)$$

Mit $G_1(z) = z * (z-1)^{-1}$ folgt:

$$E = (k_0 + k_1 * G_1) * R - k_1 * X_1 - k_2 * X_2 - (k_3 + k_1 * G_1) * X_3$$

$$G_{IF} = \frac{(k_0 + k_1 * G_1) * G_S}{1 + k_1 * G_1 + k_2 * G_1 * G_2 + (k_3 + k_1 * G_1) * G_S}$$

$$G_{IF} = \frac{(k_0(z-1) + k_1 * z) * Z_S}{(z-1) * (N_S + k_1 * Z_1 * N_2 * N_3 + k_2 * Z_1 * Z_2 * N_3) + (k_3(z-1) + k_1 * z) * Z_S}$$

$$G_{IF} = \frac{(k_0(z-1) + k_1 * z) * Z_S}{(z-1) * N_F + k_1 * z * Z_S} \quad (3.7.7.6)$$

Das Zählerpolynom $Z_{IF} = (k_0(z-1) + k_1 * z) * Z_S$ kann durch die Wahl der Konstanten k_1 und k_0 beeinflußt werden.

Für die Ermittlung des Endwertverhaltens ist die Betrachtung der

Endwert-Funktion:

$$G_E(z) = 1 - G_{IF} = \frac{(z-1) * (N_F - k_0 * Z_S)}{N_{IF}}$$

erforderlich.

Ist N_{IF} nicht durch $(z-1)$ teilbar, so wird das Endwertverhalten von der Funktion

$$Z_E = (z-1) * (N_F - k_0 * Z_S) \quad (3.7.7.7)$$

bestimmt.

Das Polynom N_{IF} ist nun genau dann durch $(z-1)$ teilbar, wenn Z_S durch $(z-1)$ teilbar ist. Dies ist äquivalent zu $Z_S(1) = 0$ und unter der Voraussetzung, daß Z_S und N_S teilerfremd sind, äquivalent zu $G_S(1) = 0$.

Aus $G_S(1) = 0$ folgt aber, daß die Sprungantwort der Strecke gegen 0 strebt. Dies kann nur bei Strecken auftreten, die einen Integralanteil besitzen, der die Strecke nach 0 ausregelt.

Im folgenden werden solche Strecken von der Betrachtung ausgeschlossen.

Es folgt dann $G_E(1) = 0$.

Die Konstante k_0 hat also keinen Einfluß auf den stationären Regelfehler. Dieser wird durch den Integralteil der Regelung in jedem Fall zu Null ausgeregelt.

Bei vorgegebenem Polynom N_{IF} kann nun k_0 dazu benutzt werden, den stationären Geschwindigkeitsfehler e_z zu Null zu machen.

Die Bedingung dafür, daß $e_z = 0$ ist, lautet:

$$N_F(1) - k_0 * Z_S(1) = 0$$

Daraus folgt wie beim P-Zustandsregler:

$$k_0 = N_F(1) / Z_S(1) \quad (3.7.7.8)$$

3.7.8 ENTWURF EINES KASKADIERTEN REGLERS

Der Entwurf eines zeitdiskreten kaskadierten Reglers kann nur bei einer Strecke mit Blockstruktur erfolgen. Der Reglerentwurf erfolgt dabei beginnend mit dem innersten Regelkreis. Beim Entwurf der überlagerten Regelkreise wird dabei jeweils die ÜF des geschlossenen unterlagerten Regelkreises zugrundegelegt. Der Reglerentwurf kann dabei nach der Methode des Zustandsreglerentwurfs oder der Methode des Einfachregelkreisentwurfs erfolgen. Werden alle digitalen Regelschleifen mit der gleichen Taktfrequenz durchlaufen, so ist die Über- und Unterlagerung nur durch den Rechengang gegeben. Die Reglerstruktur kann auch in eine Zustandsreglerstruktur umgewandelt werden. Die kaskadierte Reglerstruktur besitzt jedoch den Vorteil, daß für die Zustandsgrößen Sollwerte berechnet werden, die begrenzt werden können.

4. STROMREGLERSTRUKTUREN

4.1 ALLGEMEINES

Bei der Stromregelung kann unterschieden werden zwischen der Regelung im statorfesten System und der Regelung im rotorfesten System. Ferner kann unterschieden werden zwischen der zeitkontinuierlichen (analogen) Regelung und der zeitdiskreten (digitalen) Regelung. Bei der zeitdiskreten Regelung muß ferner unterschieden zwischen Reglerstrukturen, die die Totzeit berücksichtigen, und Reglerstrukturen, die die Totzeit vernachlässigen.

Bei der hier vorgesehenen Anwendung des geregelten Systems als Servoantrieb ist die Verwendung von Stromreglern mit I-Anteil nachteilig, da der Stromregler hier häufig an der Spannungsgrenze im nichtlinearen Bereich arbeitet. Der Integralanteil neigt dann dazu, sich unkontrolliert auf- und abzubauen. Deshalb werden in diesem Kapitel nur Stromregler ohne I-Anteil untersucht.

Die Darstellung in diesem Kapitel erfolgt in bezogenen Größen. Größen im statorfesten System werden durch ein hochgestelltes S, Größen im rotorfesten System durch ein hochgestelltes R gekennzeichnet. Fehlt diese Kennzeichnung, so geht aus dem Kontext hervor, welches System gemeint ist.

Im rotorfesten System lautet die Spannungsgleichung der PM-Synchronmaschine:

$$\underline{u} = \underline{i} \cdot (1 + j\omega) + j\omega + d\underline{i}/dt \quad (4.1.1)$$

Diese DGl kann zerlegt werden in die zwei reellen DGl:

$$u_d = i_d - \omega \cdot i_q + di_d/dt \quad (4.1.2)$$

$$u_q = i_q + \omega \cdot i_d + \omega + di_q/dt \quad (4.1.3)$$

4.2 STROMSTEUERUNG

Die Verwendung einer Stromsteuerung anstelle einer Stromregelung kann bei einer Maschine geringer Induktivität und daraus resultierender kleiner elektrischer Zeitkonstante durchaus ausreichen. Angewendet werden kann die Stromsteuerung sowohl im zeitdiskreten als auch im zeitkontinuierlichen Fall.

Unvermeidlich ist die Verwendung von Stromsteuerung, wenn die sich bei Verwendung eines Regelalgorithmus ergebende Zykluszeit zu groß ist, um die Regelung wirksam zu gestalten.

Die Verwendung einer Stromsteuerung ist insbesondere deshalb möglich, weil die Parameter der Synchronmaschine sehr gut bekannt sind. Diese Parameter sind der Strangwiderstand R , die Stranginduktivität L und die EMK-Konstante k_{EMK} .

Die Stromsteuerung benutzt die Gleichungen, die das Verhalten der Synchronmaschine bei Vorliegen des stationären Zustands im rotorfesten System beschreiben, zur Bestimmung der Sollstellgröße $\underline{u}_{s.11}$.

Die Darstellung erfolgt hier für den zeitkontinuierlichen Fall im rotorfesten System.

Die Stromsteuerung erfolgt dann nach der Gleichung:

$$\underline{u}_{s.11} = \underline{i}_{s.11} + j\omega * \underline{i}_{s.11} + j\omega \quad (4.2.1)$$

Durch Ersetzen von $\underline{u}$ in Gleichung (4.1.1) durch $\underline{u}_{s.11}$ folgt:

$$(1 + j\omega) * \underline{i}_{s.11} = (1 + j\omega) * \underline{i} + d\underline{i}/dt \quad (4.2.2)$$

$$d\underline{i}/dt = (1+j\omega) * (\underline{i}_{s.11} - \underline{i}) \quad (4.2.3)$$

Es entsteht also hier das für ein komplexes VZ1-Glied typische Übergangsverhalten (siehe/7 NAUNIN/).

Es soll nun der Einfluss von Parameterfehlern bei Verwendung der Stromsteuerung untersucht werden. Solche Parameterfehler sind unvermeidlich. Im realen Fall wird anstelle der korrekten Gleichung

(4.2.1) die fehlerhafte Gleichung:

$$\underline{u}_{s.11} = (1+a + (1+b)*jw)*\underline{i}_{s.11} + (1+c)*jw \quad (4.2.4)$$

benutzt. Die Summanden a, b und c repräsentieren dabei die relativen Parameterfehler.

Für den Stromendwert $\underline{i}_E$ bei Vorliegen des Spannungszeigers $\underline{u}_{s.11}$ gilt:

$$\underline{u}_{s.11} = (1 + jw)*\underline{i}_E + jw \quad (4.2.5)$$

Durch Gleichsetzen der Gleichungen (4.2.4) und (4.2.5) folgt:

$$(1+jw)*\underline{i}_E + jw = (1+a + (1+b)*jw)*\underline{i}_{s.11} + (1+c)*jw$$

$$(1+jw)*\underline{i}_E = (1+a + (1+b)*jw)*\underline{i}_{s.11} + c*jw$$

$$\underline{i}_E = (1+a + (1+b)*jw)/(1+jw)*\underline{i}_{s.11} + c*jw/(1+jw)$$

$$\underline{i}_E = (1 + (a+bjw)/(1+jw))*\underline{i}_{s.11} + c*jw/(1+jw)$$

$$\underline{i}_E = (1+a)*\underline{i}_{s.11} + ((b-a)*\underline{i}_{s.11} + c)*jw/(1+jw)$$

Wird keine Vorsteuerung des Stroms verwendet, so gilt:

$$\underline{i}_{s.11} = j\underline{i}_{s.11.q} \quad (4.2.6)$$

Die folgenden Überlegungen setzen die Gültigkeit der Gleichung (4.2.6) voraus.

Unter Verwendung der Identität:

$$jw/(1+jw) = w*(w+j)/(1+w^2) \quad (4.2.7)$$

folgt nun:

$$\underline{i}_{E.q} = (1+a)*\underline{i}_{s.11.q} + (b-a)*w^2/(1+w^2)*\underline{i}_{s.11.q} + c*w/(1+w^2)$$

$$\underline{i}_{E.d} = -(b-a)*w/(1+w^2)*\underline{i}_{s.11.q} + c*w^2/(1+w^2)$$

Hieraus erhält man:

$$i_{E,q} = (1+a)*is_{oll,q} + ((b-a)*w*is_{oll,q} + c)*w/(1+w^2) \quad (4.2.8)$$

$$i_{E,d} = (-(b-a)*is_{oll,q} + c*w)*w/(1+w^2) \quad (4.2.9)$$

Für $w = 0$ folgt:

$$i_{E,q} = (1+a)*is_{oll,q} \qquad i_{E,d} = 0$$

Für $|w| = 1$ folgt:

$$i_{E,q} = (1+(a+b)/2)*is_{oll,q} + c*w/2$$

$$i_{E,d} = -(b-a)*w/2*is_{oll,q} + c$$

Für $|w| \gg 1$ folgt:

$$i_{E,q} = (1+b)*is_{oll,q} \qquad i_{E,d} = c$$

Es zeigt sich hier, daß es wichtig ist, den durch die Konstante c repräsentierten Parameterfehler der EMK-Konstante klein zu halten, da dieser einen vom Sollwert $is_{oll,q}$ unabhängigen absoluten Fehler hervorruft.

Der maximale relative Stromfehler sei hier mit $i_{err,rel}$ bezeichnet:

$$i_{err,rel} = |(is_{oll,q} - i_{E,q})/is_{oll,q}|$$

Unter der Annahme $c = 0$ folgt nun:

$$w = 0 \quad : \quad i_{err,rel} \leq |a|$$

$$|w| = 1 \quad : \quad i_{err,rel} \leq |(a+b)/2|$$

$$|w| \gg 1 \quad : \quad i_{err,rel} \leq |b|$$

Der maximale relative Stromfehler wird also im wesentlichen durch die Parameterfehler a und b bestimmt. Mit $|a|, |b| < 0,1$ und $c = 0$

läßt sich also ein relativer Stromfehler von 10 % realisieren.

4.3 ZEITKONTINUIERLICHE STROMREGELUNG IM ROTORFESTEN SYSTEM

4.3.1 P-STROMREGLER OHNE STÖRGRÖßENKOMPENSATION

Hier ist wie auch in den folgenden Abschnitten der Stromistwert $\underline{i}^s$ ins rotorfeste System und der Spannungszeiger $\underline{u}^r_{s011}$ ins statorfeste System zu transformieren.

Ist ϵ der Rotorwinkel, so gilt:

$$\underline{i}^r = e^{-j\epsilon} * \underline{i}^s \quad (4.3.1.1)$$

und:

$$\underline{u}^s_{s011} = e^{j\epsilon} * \underline{u}^r_{s011} \quad (4.3.1.2)$$

Ein P-Stromregler im rotorfesten System besitzt das Reglergesetz:

$$u_{s011,d} = k_d * (i_{s011,d} - i_d) \quad (4.3.1.3)$$

$$u_{s011,q} = k_q * (i_{s011,q} - i_q)$$

Diese Regelung arbeitet nur sehr unvollkommen und ist deshalb häufig unbrauchbar. Für die Regelung der Komponenten werden hier verschiedene Reglerkonstanten benutzt, um die Regelung der q-Komponente optimieren zu können. Hier soll der bleibende Regelfehler in Abhängigkeit von w untersucht werden.

Durch Einsetzen der Modellgleichungen erhält man mit den stationären Endwerten $i_{\epsilon,d}$ und $i_{\epsilon,q}$:

$$i_{\epsilon,d} - w * i_{\epsilon,q} = k_d * (i_{s011,d} - i_{\epsilon,d})$$

$$i_{\epsilon,q} + w * i_{\epsilon,d} + w = k_q * (i_{s011,q} - i_{\epsilon,q})$$

Für $i_{E,q}$ folgt nun:

$$i_{E,q} = \frac{(k_d+1) \cdot k_q \cdot i_{soll,q} - k_d \cdot i_{soll,d} - (k_d+1) \cdot w}{(k_d+1) \cdot (k_q+1) - w^2}$$

Wird der Fall $i_{soll,d} = 0$ vorausgesetzt - also keine Vorsteuerung berücksichtigt - ,so vereinfacht sich die Gleichung zu:

$$i_{E,q} = \frac{(k_d+1) \cdot k_q \cdot i_{soll,q} - (k_d+1) \cdot w}{(k_d+1) \cdot (k_q+1) - w^2} \quad (4.3.1.4)$$

Damit der Regelfehler im gesamten Drehzahlbereich klein bleibt, muß gelten:

$$(k_d+1) \cdot (k_q+1) \gg w_{max}^2 \quad (4.3.1.5)$$

Dabei ist w_{max} die maximale Winkelgeschwindigkeit, mit der das System betrieben werden soll.

Gilt die obige Ungleichung, so kann man den Summanden w^2 im Nenner vernachlässigen und erhält:

$$i_{E,q} = \frac{k_q}{k_q+1} \cdot i_{soll,q} - \frac{w}{k_q+1} \quad (4.3.1.6)$$

Hieraus folgt, daß ein besonders kleiner Regelfehler für i_q entsteht, wenn k_q besonders groß gewählt wird.

4.3.2 P-STROMREGLER MIT KOMPENSATION DER STÖRGRÖßEN

Hier lautet das Reglergesetz im rotorfesten System:

$$u_{soll,d} = k_d \cdot (i_{soll,d} - i_d) - w \cdot i_q \quad (4.3.2.1)$$

$$u_{soll,q} = k_q \cdot (i_{soll,q} - i_q) + w \cdot i_d + w$$

Die zur Kompensation der Störgrößen dienenden Terme können im Spannungszeiger $\underline{u}_{star}$ zusammengefaßt werden:

$$\underline{u}_{\text{Stör}} = j\omega \underline{i} + j\omega \quad (4.3.2.2)$$

Auch hier soll der bleibende Regelfehler untersucht werden. Dabei muß berücksichtigt werden, daß die Berechnung der Kompensationsterme aufgrund von Parameterfehlern stets ungenau ist. Dies läßt sich berücksichtigen durch den Ansatz:

$$u_{\text{Soll,d}} = k_d * (i_{\text{Soll,d}} - i_d) - (1+a) * \omega * i_q \quad (4.3.2.3)$$

$$u_{\text{Soll,q}} = k_q * (i_{\text{Soll,q}} - i_q) + (1+a) * \omega * i_d + (1+b) * \omega$$

Dabei berücksichtigen die Summanden a und b die relativen Parameterfehler.

Eine Berechnung liefert hier im Fall $i_{\text{Soll,d}}=0$:

$$i_{\text{f,q}} = \frac{(k_d+1) * k_q * i_{\text{Soll,q}} - (k_d+1) * b * \omega}{(k_d+1) * (k_q+1) - a^2 * \omega^2} \quad (4.3.2.4)$$

Dies führt zu den leichter zu erfüllenden Bedingungen:

$$(k_d+1) * (k_q+1) \gg a^2 * \omega_{\text{max}}^2$$

$$k_q+1 \gg b * \omega_{\text{max}}$$

Anstelle der Kompensation mit den Istwerten kann die Kompensation auch mit den Sollwerten durchgeführt werden.

Dies führt zu dem Reglergesetz:

$$u_{\text{Soll,d}} = k_d * (i_{\text{Soll,d}} - i_d) - \omega * i_{\text{Soll,q}} \quad (4.3.2.5)$$

$$u_{\text{Soll,q}} = k_q * (i_{\text{Soll,q}} - i_q) + \omega * i_{\text{Soll,d}} + \omega$$

4.3.3 P-STROMZUSTANDSREGLER

Bei gleichen Reglerkonstanten für die d- und q-Komponente lautet das Reglergesetz:

$$u_{s011,d} = k_1 * i_{s011,d} - k_2 * i_d - w * i_q \quad (4.3.3.1)$$

$$u_{s011,q} = k_1 * i_{s011,q} - k_2 * i_q + w * i_d + w$$

Der Zustandsregler besitzt bessere Eigenschaften als der in Abschnitt (4.3.2) beschriebene P-Regler.

Wird der Regler als DEAD-BEAT-Regler entworfen, so gilt:

$$k_1 = k_2 + 1 \quad (4.3.3.2)$$

Gleichung (4.3.3.3) kann dann umgeformt werden in:

$$u_{s011,d} = k_1 * (i_{s011,d} - i_d) + i_d - w * i_q \quad (4.3.3.3)$$

$$u_{s011,q} = k_1 * (i_{s011,q} - i_q) + i_q + w * i_d + w$$

Der so entstandene Regler kann auch interpretiert werden als P-Regler, der von einem um den jeweiligen Arbeitspunkt linearisierten Modell ausgeht.

4.4 ZEITDISKRETE STROMREGELUNG OHNE TOTZEIT

4.4.1 ALLGEMEINES

In diesem Abschnitt (4.4) wird der Fall betrachtet, daß durch die Rechenzeit bedingte Totzeit gegenüber der Reglertaktzeit T vernachlässigbar ist.

Im folgenden treten Schätzwerte für Zustandsgrößen auf. Diese Schätzwerte werden wie zur Unterscheidung von den Meßwerten mit einem hochgestellten B gekennzeichnet.

Bezüglich der Indizierung der zeitdiskreten Werte wird in diesem Abschnitt folgende Vereinbarung getroffen:

1. Bei der Stellgröße $\underline{u}$ wird der Anfangszeitpunkt des Intervalls, in dem $\underline{u}$ wirkt, zur Indizierung verwendet.
2. Bei gemessenen Größen wird der Zeitpunkt, zu dem die Messung erfolgte, zur Indizierung verwendet. Es wird angenommen, daß der Zeitpunkt der Messung mit dem Zeitpunkt der Änderung der Stellgröße identisch ist.

4.4.2 STROMZUSTANDSREGLER IM ROTORFESTEN SYSTEM

Bei der Stromregelung im rotorfesten System ist der Stromistwert $\underline{i}_n^s$ in das rotorfeste System und der Spannungszeiger $\underline{u}_{soll,n}^R$ ins statorfeste System umzurechnen. Die Umrechnung erfolgt gemäß den Formeln:

$$\underline{i}_n^R = \underline{i}_n^s \cdot \exp(-j\varepsilon_n) \quad (4.4.2.1)$$

$$\underline{u}_{soll,n}^S = \underline{u}_{soll,n}^R \cdot \exp(j\varepsilon_{n+1/2}) \quad (4.4.2.2)$$

Die Bestimmung des Winkels $\varepsilon_{n+1/2}$ kann erfolgen gemäß:

$$\varepsilon_{n+1/2} = \varepsilon_n + T/2 \cdot \omega_n \quad (4.4.2.3)$$

Das Gesetz des Stromzustandsreglers im rotorfesten System lautet dann:

$$\underline{u}^{*R}_{soll,n} = k_1 \cdot \underline{i}_{soll,n}^R - k_2 \cdot \underline{i}_n^R \quad (4.4.2.4)$$

Mit

$$\underline{u}^{R}_{stör,n} = j\omega_n \cdot \underline{i}_n^R + j\omega_n \quad (4.4.2.5)$$

erfolgt die Berechnung der Stellgröße $\underline{u}_{soll,n}^R$:

$$\underline{u}_{soll,n}^R = \underline{u}^{*R}_{soll,n} + \underline{u}^R_{stör,n} \quad (4.4.2.6)$$

Durch Einsetzen folgt:

$$\underline{u}^{R_{soll,n}} = k_1 * \underline{i}^{R_{soll,n}} - k_2 * \underline{i}^R_n + j\omega_n * \underline{i}^R_n + j\omega_n \quad (4.4.2.7)$$

Die Formel (4.4.2.5) ist dem zeitkontinuierlichen Reglergesetz entnommen. Die Verwendung dieser Formel bringt eine quasikontinuierliche Betrachtungsweise in das Reglergesetz ein.

Bei Vorliegen sehr kleiner Regeltaktzeiten gemessen an der elektrischen Zeitkonstante des Systems ist die Kompensation mit Stromistwerten vorzuziehen, da die Komensation um so wirksamer ist, je genauer der verwendete Stromwert mit dem tatsächlichen Stromwert übereinstimmt.

Bei Vorliegen relativ großer Regeltaktzeiten kann die Verwendung der Kompensation mit Stromsollwerten vorteilhaft sein, da sie im Sinne einer Steuerung die Regelung unterstützt. Dies führt zu der Gleichung:

$$\underline{u}^{R_{stör,n}} = j\omega_n * \underline{i}^{R_{soll,n}} + j\omega_n \quad (4.4.2.8)$$

Eine Verbesserung des Systemverhaltens bei Kompensation mit Sollwerten anstelle der Kompensation mit Istwerten ist z.B. von /9 ORLIK/ bei Verwendung eines PI-Reglers festgestellt worden.

4.4.3 STROMZUSTANDSREGLER IM STATORFESTEN SYSTEM

Bei der Stromregelung im statorfesten System sind der Stromsollwert $\underline{i}^{R_{soll,n}} = \underline{i}_{soll,n}$ und der Polradspannungszeiger $\underline{u}_p^R_n = j\omega_n$ ins statorfeste System umzurechnen. Die Umrechnung erfolgt gemäß den Formeln:

$$\underline{i}^S_{soll,n} = \underline{i}^R_{soll,n} * \exp(j\varepsilon^B_{n+1}) \quad (4.4.3.1)$$

$$\underline{u}_p^S_n = j\omega_n * \exp(j\varepsilon^B_{n+1/2}) \quad (4.4.3.2)$$

Dabei kann gesetzt werden:

$$\varepsilon^B_{n+1} = \varepsilon_n + T * \omega_n \quad (4.4.3.3)$$

$$\varepsilon_{n+1/2}^B = \varepsilon_n + T/2 * \omega_n \quad (4.4.3.4)$$

Das Gesetz des Stromzustandsreglers im statorfesten System lautet dann:

$$\underline{u}_{soll,n}^S = k_1 * \underline{i}_{soll,n}^S - k_2 * \underline{i}_n^S + \underline{u}_p^S \quad (4.4.3.5)$$

Verwendet man anstelle von (4.4.3.4) die nicht ganz korrekte Gleichung (4.4.3.3) zur Berechnung von $\varepsilon_{n+1/2}^B$, so kommt man bei der Berechnung der Stellgröße mit einer Winkeltransformation aus:

$$\underline{u}_{soll,n}^S = (k_1 * \underline{i}_{soll,n}^R + j\omega_n) * e^{j\varepsilon_n^B} - k_2 * \underline{i}_n^S \quad (4.4.3.6)$$

4.4.4 VERGLEICH DER REGLERSTRUKTUREN

Der während einer Reglertaktzeit geschaltete Spannungszeiger ist stets ein statorfester Spannungszeiger. Dieser wird bei Verwendung des 3-Vektor-Verfahrens aus drei statorfesten Spannungszeigern zusammengesetzt. Dies zeigt, daß bei der Berechnung die Verwendung des statorfesten Systems der natürliche Weg ist.

Es soll nun das Reglergesetz des in Abschnitt (4.4.2) beschriebenen rotorfesten Reglers ins statorfeste System umgerechnet werden.

Aus Reglergesetz (4.4.2.7) erhält man mit (4.4.2.2) :

$$\underline{u}_{soll,n}^S = (k_1 * \underline{i}_{soll,n}^R - k_2 * \underline{i}_n^R + j\omega_n * \underline{i}_n^R + j\omega_n) * \exp(j\varepsilon_{n+1/2}^B)$$

Unter Verwendung von (4.4.2.1) folgt nun:

$$\begin{aligned} \underline{u}_{soll,n}^S &= k_1 * \exp(j\varepsilon_{n+1/2}^B) * \underline{i}_{soll,n}^R - \\ &- (k_2 - j\omega_n) * \exp(j\omega_n * T/2) * \underline{i}_n^S + \\ &+ j\omega_n * \exp(j\varepsilon_{n+1/2}^B) \end{aligned}$$

Mit (4.4.3.1) und (4.4.3.2) erhält man:

$$\begin{aligned} \underline{u}_{soll,n} &= k_1 \cdot \exp(-j \cdot \omega_n \cdot T/2) \cdot \underline{i}_{soll,n} - \\ &- (k_2 - j \omega_n) \cdot \exp(j \cdot \omega_n \cdot T/2) \cdot \underline{i}_n + \\ &+ \underline{u}_p^n \end{aligned} \quad (4.4.4.1)$$

Für $\omega_n = 0$ ist also das Reglergesetz im rotorfesten System identisch mit dem Reglergesetz (4.4.3.5) im statorfesten System. Dies gilt auch näherungsweise, wenn $|\omega_n| \ll 1$ ist. Ist diese Bedingung nicht erfüllt, so ergeben sich größere Abweichungen zwischen den Reglergesetzen. Dabei muß das Reglergesetz im statorfesten System als das exaktere betrachtet werden.

Im folgenden wird eine Umformung der Gleichung (4.4.4.1) für den Fall $|\omega_n \cdot T/2| \ll 1$ gegeben.

Für $|x| \ll 1$ gilt:

$$\exp(jx) = \cos(x) + j \cdot \sin(x) \approx 1 + jx$$

Unter der Bedingung $|\omega_n \cdot T/2| \ll 1$ folgt also:

$$\begin{aligned} \underline{u}_{soll,n} &= k_1 \cdot (1 - j \omega_n T/2) \cdot \underline{i}_{soll,n} - \\ &- (k_2 + \omega_n^2 \cdot T/2 + j \cdot (-\omega_n + k_2 \cdot \omega_n \cdot T/2)) \cdot \underline{i}_n + \\ &+ \underline{u}_p^n \end{aligned}$$

Hieraus erhält man:

$$\begin{aligned} \underline{u}_{soll,n} &= k_1 \cdot \underline{i}_{soll,n} - k_2 \cdot \underline{i}_n + \underline{u}_p^n - \\ &- j \omega_n \cdot k_1 \cdot T/2 \cdot \underline{i}_{soll,n} + \\ &- (\omega_n^2 \cdot T/2 + j \omega_n \cdot (-1 + k_2 \cdot T/2)) \cdot \underline{i}_n \end{aligned} \quad (4.4.4.2)$$

Zu den Summanden des statorfesten Reglers (4.4.3.5) treten also

unter den gemachten Voraussetzungen bei Verwendung des rotorfesten Reglers (4.4.2.7) zwei weitere Summanden hinzu.

Die Regelung im rotorfesten System besitzt gegenüber der Regelung im statorfesten System den Vorteil, daß hier die Größen i_d und i_q berechnet werden. So ist es hier möglich, bei der Regelung der Komponenten verschiedene Konstanten zu verwenden. Eine ungleiche Behandlung der Komponenten im statorfesten System ist jedoch nicht sinnvoll.

4.4.5 STROMZUSTANDSREGLER IM GEDREHTEN STATORFESTEN SYSTEM

Die Vorteile der in den Abschnitten 4.4.3 und 4.4.4 beschriebenen Regler können kombiniert werden, indem ein statorfestes System verwendet wird, das zum Zeitpunkt $(n+1/2)*T$ mit dem rotorfesten System übereinstimmt.

Die Größen in diesem System werden durch ein hochgestelltes D gekennzeichnet. Die Regelung in diesem System hat gegenüber der Regelung im statorfesten xy-System den Vorteil, daß die Komponenten des Stromzeigers mit unterschiedlichen Konstanten geregelt werden können.

Die Transformationen erfolgen mit dem Winkel:

$$\varepsilon^{B_{n+1/2}} = \varepsilon_n + \omega_n * T/2 \quad (4.4.5.1)$$

gemäß:

$$\underline{i}^D_n = \underline{i}^S_n * \exp(-j\varepsilon^{B_{n+1/2}}) \quad (4.4.5.2)$$

$$\underline{i}^{D_{soll,n}} = \underline{i}^{R_{soll,n}} * \exp(j\omega_n * T/2) \quad (4.4.5.3)$$

$$\underline{u}^S_{soll,n} = \underline{u}^D_{soll,n} * \exp(j\varepsilon^{B_{n+1/2}}) \quad (4.4.5.4)$$

Das Gesetz des Stromzustandsreglers im gedrehten statorfesten System lautet dann:

$$\underline{u}^D_{soll,n} = k_1 * \underline{i}^D_{soll,n} - k_2 * \underline{i}^D_n + j\omega_n \quad (4.4.5.5)$$

Der Rechenaufwand vergrößert sich gegenüber der Berechnung im statorfesten System, da drei Transformationen erforderlich sind.

4.5 STROMREGLERSTRUKTUREN MIT TOTZEITBERÜCKSICHTIGUNG

4.5.1 ALLGEMEINES

Bei Verwendung eines digitalen Reglers ist das Auftreten einer Totzeit unvermeidlich. Unter Totzeit T_r wird die Zeit verstanden, die zwischen der Erfassung der Messwerte und dem Wirksamwerden der Stellgröße vergeht. Diese Totzeit setzt sich zusammen aus der Reglertotzeit und der Totzeit der Stelleinrichtung, d.h. hier des Pulswechselrichters. Die Totzeit des Wechselrichters kann im vorliegenden Fall gegenüber der Reglertotzeit vernachlässigt werden.

Die Reglertaktzeit T ist bei vorgebenem Regelalgorithmus und vorgegebener Hardware bestimmt durch die zur Ausführung des Regelalgorithmus notwendige Zeit T_{r0} :

$$T \geq T_{r0}$$

Bei einem Einprozessorsystem muß $T > T_{r0}$ sein, da zum Betrieb des Hintergrundprogramms, das u.a. für die Ein- und Ausgabe von Daten an andere Systemeinheiten zuständig ist, Rechenzeit benötigt wird.

Beim hier vorliegenden System ist die Totzeit T_r gleich der Taktzeit T :

$$T_r = T$$

Das Zeitablaufdiagramm der digitalen Regelung ist in Bild 4.1 dargestellt. Es wird dabei davon ausgegangen, daß die Zeitpunkte, zu denen Messungen erfolgen, mit den Zeitpunkten, in denen Änderungen der Stellgröße erfolgen, zusammenfallen.

Bezüglich der Indizierung der zeitdiskreten Werte wird hier folgende Vereinbarung getroffen:

1. Bei der Stellgröße u wird der Anfangszeitpunkt des Intervalls, in dem u wirkt, zur Indizierung verwendet.

2. Bei gemessenen Größen wird der Zeitpunkt, zu dem die Messung erfolgte, zur Indizierung verwendet. Es wird angenommen, daß der Zeitpunkt der Messung mit dem Anfangszeitpunkt des Regelalgorithmus identisch ist.

3. Bei Größen, die nur in digitaler Form im Regelalgorithmus auftreten, wird der Anfangszeitpunkt des Regelalgorithmus verwendet. Dies gilt auch für den von einem übergeordneten System gelieferten Sollwert ε_{soll} bzw. w_{soll} .

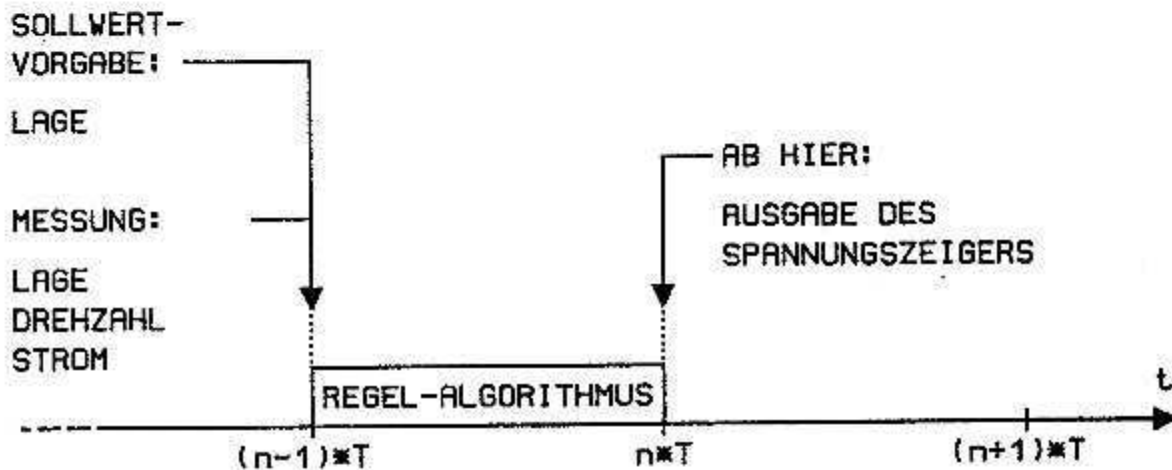


Bild 4.1 Zeitablaufdiagramm der digitalen Regelung

4.5.2 STROMZUSTANDSREGLER IM ROTORFESTEN SYSTEM

Hier ist der Stromistwert $\underline{i}_n^s$ in das rotorfeste System und der Spannungszeiger $\underline{u}_{soll,n+1}^r$ ins statorfeste System umzurechnen. Die Umrechnung erfolgt gemäß den Formeln:

$$\underline{i}_n^r = \underline{i}_n^s \cdot \exp(-j\varepsilon_n) \quad (4.5.2.1)$$

$$\underline{u}_{soll,n}^s = \underline{u}_{soll,n}^r \cdot \exp(j\varepsilon_{n+3/2}) \quad (4.5.2.2)$$

Die Bestimmung des Winkels $\varepsilon_{n+3/2}$ kann erfolgen gemäß:

$$\varepsilon_{n+3/2} = \varepsilon_n + \omega_n \cdot 3T/2 \quad (4.5.2.3)$$

Aus dem zeitdiskreten Übergangsmodell folgt mit $a = e^{-T}$:

$$\underline{i}^{B,R}_{n+1} = a * \underline{i}^R_n + (1-a) * \underline{u}^{*R}_n \quad (4.5.2.4)$$

Es wird nun der Regleransatz gemacht:

$$\underline{u}^{*R}_{soll,n+1} = k_1 * \underline{i}^R_{soll,n} - k_2 * \underline{i}^R_{n+1} \quad (4.5.2.5)$$

Ersetzen des unbekannten Werts $\underline{i}^R_{n+1}$ durch $\underline{i}^{B,R}_{n+1}$ liefert:

$$\underline{u}^{*R}_{soll,n+1} = k_1 * \underline{i}^R_{soll,n} - k_2 * a^2 * \underline{i}^R_n - k_2 * (1-a) * \underline{u}^{*R}_n \quad (4.5.2.6)$$

Die Gleichung für die Kompensation lautet:

$$\underline{u}^R_{soll,n+1} = \underline{u}^{*R}_{soll,n+1} + \underline{u}^R_{stör,n+1} \quad (4.5.2.7)$$

Bei der Berechnung von $\underline{u}^R_{stör,n+1}$ wird die Totzeit vernachlässigt, sodaß gilt:

$$\underline{u}^R_{stör,n+1} = jw_n * \underline{i}^R_n + jw_n \quad (4.5.2.8)$$

Der Wert $\underline{u}^{*R}_n$ ist dabei zu bestimmen gemäß:

$$\underline{u}^{*R}_n = \underline{u}^R_n - \underline{u}^R_{stör,n} \quad (4.5.2.9)$$

Ist also $\underline{u}^R_n = \underline{u}^R_{soll,n}$, so gilt $\underline{u}^{*R}_n = \underline{u}^{*R}_{soll,n}$.

Werden die Reglerkonstanten k_1 und k_2 direkt aus dem Übergangsmodell abgeleitet, so folgt:

$$k_1 = 1/(1-a) \quad k_2 = a/(1-a) \quad (4.5.2.10)$$

Der Regler erhält damit DEAD-BEAT-Verhalten. Dies bedeutet in diesem Fall, daß der Sollwert nach genau zwei Taktperioden erreicht wird.

Mit (4.5.2.10) erhält das Reglergesetz (4.5.2.6) die Form:

$$\underline{u}^{*R}_{soll,n+1} = 1/(1-a) * \underline{i}^R_{soll,n} - a^2/(1-a) * \underline{i}^R_n - a * \underline{u}^{*R}_n \quad (4.5.2.11)$$

4.5.3 STROMZUSTANDSREGLER IM ROTORFESTEN SYSTEM MIT ZWISCHENZUSTANDSBERECHNUNG

Soll die Totzeit auch bei der Kompensation berücksichtigt werden, so muß der Zwischenzustand $\underline{i}^{B \cdot R}_{n+1}$ gemäß Gleichung (4.5.2.4) berechnet werden:

$$\underline{i}^{B \cdot R}_{n+1} = a * \underline{i}^{R_n} + (1-a) * \underline{u}^{R_n} \quad (4.5.3.1)$$

Dieser Zwischenzustand kann dann sowohl bei der Regelung als auch bei der Kompensation verwendet werden.

Das Reglergesetz des Stromzustandsreglers vereinfacht sich dann zu:

$$\underline{u}^{R_{soll, n+1}} = k_1 * \underline{i}^{R_{soll, n}} - k_2 * \underline{i}^{B \cdot R}_{n+1} \quad (4.5.3.2)$$

Die Berechnung des Kompensationszeigers erfolgt nach der Formel:

$$\underline{u}^{R_{stör, n+1}} = j\omega_n * \underline{i}^{B \cdot R}_{n+1} + j\omega_n \quad (4.5.3.3)$$

In dieser Gleichung wird die Totzeit nur noch bei der Drehzahl vernachlässigt.

4.5.4 STROMZUSTANDSREGLER IM STATORFESTEN SYSTEM

Hier sind der Stromsollwert $\underline{i}^{R_{soll, n}}$ und der Polradspannungszeiger $\underline{u}^{R_{n+1}} \approx j\omega_n$ ins statorfeste System umzurechnen. Die Umrechnung erfolgt gemäß den Formeln:

$$\underline{i}^{S_{soll, n}} = \underline{i}^{R_{soll, n}} * \exp(j\varepsilon^{B}_{n+2}) \quad (4.5.4.1)$$

$$\underline{u}^{S_{n+1}} = j\omega_n * \exp(j\varepsilon^{B}_{n+3/2}) \quad (4.5.4.2)$$

Dabei kann gesetzt werden:

$$\varepsilon^{B}_{n+2} = \varepsilon_n + 2\omega_n T \quad (4.5.4.3)$$

$$\varepsilon^{B}_{n+3/2} = \varepsilon_n + \omega_n * 3T/2 \quad (4.5.4.4)$$

Mit

$$\underline{u}^{*s_n} = \underline{u}^{s_n} - \underline{u}_p^{s_n}$$

liefert das zeitdiskrete Modell wie im rotorfesten System:

$$\underline{i}^{s_{n+1}} = a * \underline{i}^{s_n} + (1-a) * \underline{u}^{*s_n} \quad (4.5.4.5)$$

Mit dem Regleransatz:

$$\underline{u}^{*s_{soll,n+1}} = k_1 * \underline{i}^{s_{soll,n}} - k_2 * \underline{i}^{s_{n+1}} \quad (4.5.4.6)$$

folgt ebenfalls wie im rotorfesten System:

$$\underline{u}^{*s_{soll,n+1}} = k_1 * \underline{i}^{s_{soll,n}} - k_2 * (a * \underline{i}^{s_n} + (1-a) * \underline{u}^{*s_n}) \quad (4.5.4.7)$$

Mit

$$\underline{u}^{s_{soll,n+1}} = \underline{u}^{*s_{soll,n+1}} + \underline{u}_p^{s_{n+1}} \quad (4.5.4.8)$$

folgt:

$$\underline{u}^{s_{soll,n+1}} = k_1 * \underline{i}^{s_{soll,n}} - k_2 * a * \underline{i}^{s_n} - k_2 * (1-a) * \underline{u}^{*s_n} + \underline{u}_p^{s_{n+1}} \quad (4.5.4.9)$$

Der Regler besitzt DEAD-BEAT-Verhalten, wenn gilt:

$$\underline{u}^{*s_{soll,n+1}} = 1/(1-a) * \underline{i}^{s_{soll,n}} - a^2/(1-a) * \underline{i}^{s_n} - a * \underline{u}^{*s_n} + \underline{u}_p^{s_{n+1}} \quad (4.5.4.10)$$

4.5.5 VERGLEICH DER STROMZUSTANDSREGLER

Bei den hier beschriebenen Stromreglern mit Berücksichtigung der Totzeit sind stets zwei Transformationen erforderlich.

Bei Stromregelung im rotorfesten System ist der Stromistwert ins rotorfeste System und der berechnete Spannungszeiger ins statorfeste System zu transformieren.

Bei Stromregelung im statorfesten System ist der Stromsollwert und die Polradspannung ins statorfeste System zu transformieren.

Der in Abschnitt 4.5.2 beschriebene Stromregler zeigt ein geringfügig schlechteres dynamisches Verhalten als die in den Abschnitten 4.5.3 und 4.5.4 beschriebenen Regler. Die beiden letztgenannten Regler zeigen gleiches Verhalten.

Das Verhalten verschiedener Stromregler wurde sowohl simuliert als auch durch Messungen am realen System bestimmt. Simulationsergebnisse und Messergebnisse zum Reglerverhalten finden sich in einem späteren Kapitel.

5. REGLERSTRUKTUREN

5.1 ALLGEMEINES

In diesem Kapitel sollen verschiedene Reglerstrukturen miteinander verglichen werden. Dabei erweisen sich sehr unterschiedliche Reglerstrukturen als äquivalent. Die Äquivalenz erweist sich dadurch, daß es möglich ist, die Reglergesetze ineinander zu überführen.

Zu den Eingangsgrößen der Regler gehören stets die Lage ε und die Drehzahl w . Die Eingangsgröße w kann dabei durch Messung mit einem Tachogenerator oder wie im hier betrachteten System durch Berechnung aus den zeitdiskreten Lagewerten ε gemäß der Formel:

$$w_n = \frac{\varepsilon_n - \varepsilon_{n-1}}{T} \quad (5.1.1)$$

erfolgen.

In beiden Fällen wird die Gültigkeit der Beziehung:

$$w = \frac{d\varepsilon}{dt} \quad (5.1.2)$$

vorausgesetzt. Dies bedeutet, daß Unvollkommenheiten des Tachogenerators bzw. die durch Verwendung der Formel (5.1.1) bedingten Abweichungen nicht berücksichtigt werden.

Bei der Umformung der Reglergesetze kann nun die aus (5.1.2) folgende Gleichung:

$$\int_0^t w \, d\tau = \varepsilon(t) - \varepsilon(0) \quad (5.1.3)$$

benutzt werden.

Im zeitdiskreten Fall erhält man unter Verwendung von (5.1.1):

$$\int_0^T w \, d\tau = T * \sum_{i=1}^n w_n = \varepsilon_n - \varepsilon_0 \quad (5.1.4)$$

Die Gleichung (5.1.3) behält also auch im zeitdiskreten Fall ihre Gültigkeit.

5.2 REGLERSTRUKTUREN OHNE BERÜCKSICHTIGUNG DER TOTZEIT

5.2.1 KLASSISCHE KASKADENREGLERSTRUKTUR

Die Darstellung in den folgenden Abschnitten erfolgt zeitkontinuierlich, da diese Darstellung zu übersichtlicheren Gleichungen führt. Die gemachten Aussagen sind jedoch auch für den zeitdiskreten Fall gültig. Die klassische Kaskadenreglerstruktur zur Lageregelung mit elektrischen Antrieben enthält einen P-Lageregler und einen unterlagerten PI-Drehzahl- und PI-Stromregler. Bild 5.1 zeigt diese Reglerstruktur. Die Verwendung eines Drehzahlreglers mit I-Anteil anstelle eines denkbaren Lagereglers mit I-Anteil ist sinnvoll, weil das als Störgröße wirkende Lastmoment m_l innerhalb der Drehzahlregelschleife angreift.

Der Stromregelkreis ist stets maschinentypabhängig. Er enthält i.A. Terme zur Kompensation von Störgrößen.

Bei der Wahl eines Stromreglers kann auf die in Kapitel 4 beschriebenen Stromreglerstrukturen zurückgegriffen werden.

Die klassische Reglerstruktur ist dadurch gekennzeichnet, daß sämtliche Zustandsgrößen in die Regelung eingehen.

Das Reglergesetz des P-Lagereglers lautet:

$$w_{soll} = k_1 * (\varepsilon_{soll} - \varepsilon) \quad (5.2.1.1)$$

Das Reglergesetz des PI-Drehzahlreglers lautet:

$$i_{q,soll} = k_2 * (w_{soll} - w) + k_1 * (w_{soll} - w) \, dt \quad (5.2.1.2)$$

$$i_{d,soll} = 0$$

Bei Verwendung des in 4.3.2 beschriebenen P-Stromreglers gilt:

$$\underline{u}^*_{soll} = k_3 * (\underline{i}_{soll} - \underline{i}) \quad (5.2.1.3)$$

Die Stellgröße $\underline{u}_{soll}$ entsteht hieraus durch Addition des Kompensationsterms $\underline{u}_{komp}$:

$$\underline{u}_{soll} = \underline{u}_{komp} + \underline{u}^*_{soll} \quad (5.2.1.4)$$

$$\underline{u}_{komp} = -W * \underline{i}_q + j * (W * \underline{i}_d + W) \quad (5.2.1.5)$$

Die Gleichungen (5.2.1.1) bis (5.2.1.5) beschreiben somit den kaskadierten Lageregler.

Verwendet man anstelle des bei diesem System unzulänglichen P-Stromreglers den in Abschnitt 4.3.3 beschriebenen Strom-Zustandsregler, so ist die Gleichung (5.2.1.3) zu ersetzen durch:

$$\underline{u}^*_{soll} = k_3 * \underline{i}_{soll} - k_4 * \underline{i} \quad (5.2.1.6)$$

Durch Umformung des Reglergesetzes läßt sich die dann entstehende Reglerstruktur in die konventionelle Kaskadenreglerstruktur einpassen.

Hierzu wird ein 'Stromsollwert' $\underline{i}^*_{soll}$ definiert gemäß:

$$\underline{i}^*_{soll} = k_3 / k_4 * \underline{i}_{soll} \quad (5.2.1.7)$$

Damit erhält man aus (5.2.1.6):

$$\underline{u}^*_{soll} = k_4 * (\underline{i}^*_{soll} - \underline{i}) \quad (5.2.1.8)$$

Gleichung (5.2.1.2) kann nun umgeformt werden in:

$$\dot{\underline{i}}^*_{q,soll} = k_2 * (w_{soll} - w) + k_1 * \int (w_{soll} - w) dt \quad (5.2.1.8)$$

, wobei gilt: $k_2^* = k_2 * k_3 / k_4$ und $k_1^* = k_1 * k_3 / k_4$

Diese Umformung vermindert den Rechenaufwand um eine Multiplikation. Wird eine Begrenzung von $\underline{i}^*_{soll}$ durchgeführt, so ist

zu berücksichtigen, daß i^*_{soll} nicht der Sollwert, sondern eine dem Sollwert gemäß (5.2.1.6) proportionale Größe ist.

Zur Berechnung der Stellgröße u^*_{soll} nach den Gleichungen (5.2.1.1) bis (5.2.1.3) sind 5 Multiplikationen erforderlich. Zur Berechnung des Kompensationsterms u_{komp} sind unter Berücksichtigung der Tatsache, daß die Berechnung nicht in bezogenen Größen durchgeführt werden kann, zusätzlich 4 Multiplikationen erforderlich.

5.2.2 Zustandsreglerstruktur

Eine reine Zustandsreglerstruktur mit Integralanteil führt zum Reglergesetz:

$$u^*_{d.soll} = -h_3 * i_d \quad (5.2.2.1)$$

$$u^*_{e.soll} = h_0 * \varepsilon_{soll} - h_1 * \varepsilon - h_2 * w - h_3 * i_q + h_1 * (\varepsilon_{soll} - \varepsilon) dt \quad (5.2.2.2)$$

Die Stellgröße u_{soll} entsteht wieder gemäß den Gleichungen (5.2.1.4) und (5.2.1.5). Die Reglerkonstanten des Zustandsreglers sind hier zur Unterscheidung von denen des kaskadierten Reglers mit dem Buchstaben h bezeichnet. Die auftretenden Reglerkonstanten sind hierbei sämtlich positiv:

$$h_0, h_1, h_2, h_3, h_1 > 0$$

Das Reglergesetz des Zustandsreglers muß invariant gegenüber Lageverschiebungen sein. Beim Kaskadenregler ist diese Invarianz gegeben, da dort nur die Lagedifferenz $\varepsilon_{soll} - \varepsilon$ in das Reglergesetz eingeht. Ist $h_0 = h_1$, so ist das Gesetz (5.2.2.2) nicht invariant gegenüber Lageverschiebungen.

Diese Invarianz kann dadurch erreicht werden, daß zum Reglergesetz der additive Term $(h_1 - h_0) * \varepsilon_0$ hinzuzufügt wird, wobei ε_0 die Lage zum Zeitpunkt $t = 0$ ist:

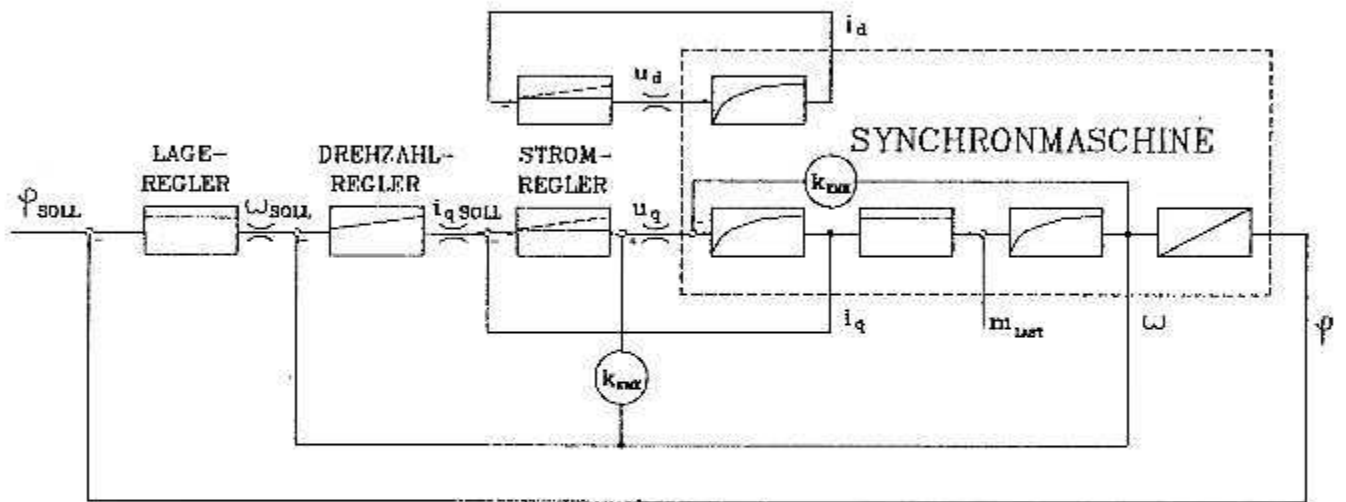


Bild 5.1 Klassische Kaskadenreglerstruktur

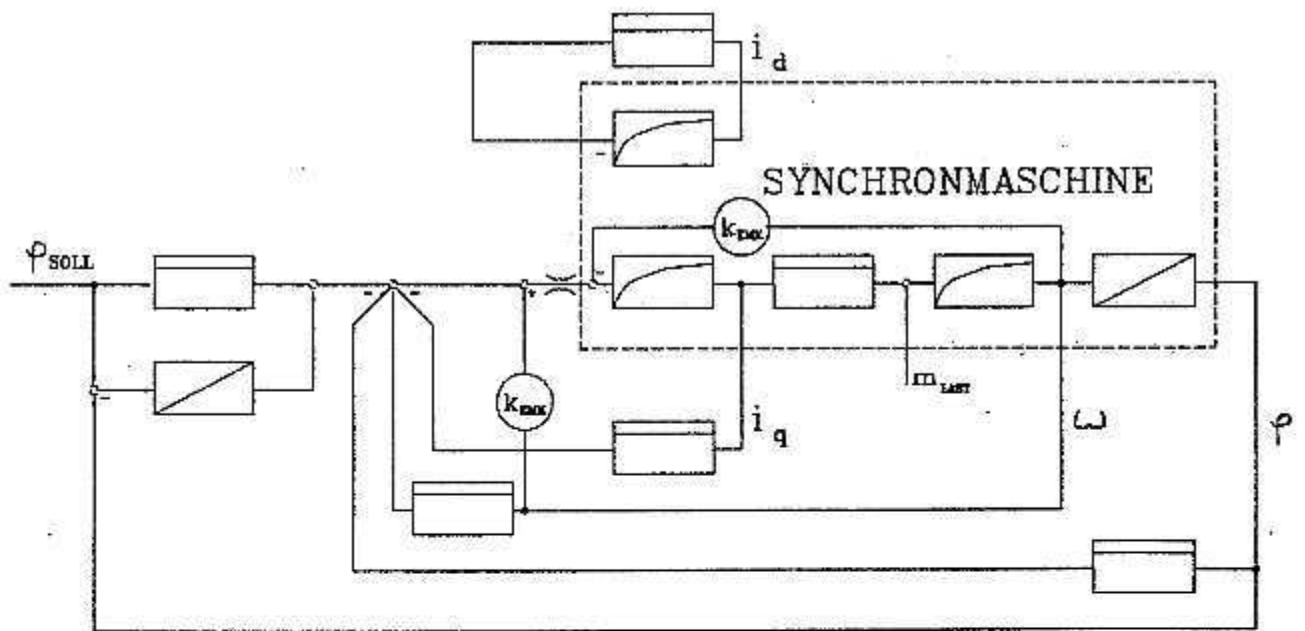


Bild 5.2 Lageinvarianter Zustandsregler mit Störgrößenkompensation

reicht ist, gelten:

$$\varepsilon = \varepsilon_{soll}, w = 0, i_q = 0, u_{q,soll}^* = 0$$

Aus dem Reglergesetz (5.2.2.4) erhält man dann:

$$(h_1 - h_0) * (\varepsilon_{soll} - \varepsilon_0) = h_1 * (\varepsilon_{soll} - \varepsilon) dt \quad (5.2.2.7)$$

Aus dieser Gleichung folgt unmittelbar, daß der Regler überschwingen muß, wenn $h_0 = h_1$ gewählt wird.

Für einen Regler, der die Sollage im Fall $m_1 = 0$ ohne Überschwingen erreicht, muß gelten $h_1 > h_0$.

Zur Berechnung der Stellgröße u_{soll}^* nach dem Zustandsreglergesetz (5.2.2.4) sind wie beim kaskadierten Regler 5 Multiplikationen erforderlich.

Die Anzahl der insgesamt erforderlichen Multiplikationen kann um eine auf 8 verringert werden, indem die Kompensation der Polradspannung in das Zustandsreglergesetz aufgenommen wird:

$$u_{d,soll}^* = -h_3 * i_d$$

$$u_{q,soll}^* = h_1 * (\varepsilon_{soll} - \varepsilon) + (h_0 - h_1) * (\varepsilon_{soll} - \varepsilon_0) - (h_2 - 1) * w - h_3 * i_q + h_1 * (\varepsilon_{soll} - \varepsilon) dt \quad (5.2.2.8)$$

$$u_{soll} = u_{soll}^* + jw * i \quad (5.2.2.9)$$

5.2.3 Vergleich der Reglerstrukturen

Zum Vergleich der Reglerstrukturen wird das Reglergesetz des kaskadierten Reglers in die Zustandsreglerstruktur überführt. Hierzu werden aus dem Reglergesetz (5.2.1.1) bis (5.2.1.3) die Zwischensollwerte w_{soll} und $i_{q,soll}$ eliminiert.

Durch Einsetzen von (5.2.1.1) in (5.2.1.2) folgt:

$$i_{q,soll} = k_1 * k_2 * (\varepsilon_{soll} - \varepsilon) - k_2 * w + k_1 * k_1 * (\varepsilon_{soll} - \varepsilon) dt - k_1 * w dt$$

Durch Verwendung der Gleichung (5.1.3) erhält man:

$$\dot{i}_{q, \text{ Soll}} = k_1 * k_2 * (\varepsilon_{\text{Soll}} - \varepsilon) - k_2 * w + k_1 * k_i * (\varepsilon_{\text{Soll}} - \varepsilon) dt - k_i * (\varepsilon - \varepsilon_0)$$

Hieraus folgt:

$$\begin{aligned} i_{q, \text{ Soll}} = & k_1 * k_2 * \varepsilon_{\text{Soll}} - (k_1 * k_2 + k_i) * \varepsilon - k_2 * w + k_1 * \varepsilon_0 + \\ & + k_i * k_1 * (\varepsilon_{\text{Soll}} - \varepsilon) dt \end{aligned}$$

Durch Einsetzen dieser Gleichung in (5.2.1.3) folgt:

$$\begin{aligned} u^*_{q, \text{ Soll}} = & h_0 * \varepsilon_{\text{Soll}} - h_1 * \varepsilon + (h_1 - h_0) * \varepsilon_0 - h_2 * w - h_3 * i_q + \\ & + h_i * (\varepsilon_{\text{Soll}} - \varepsilon) dt \end{aligned} \quad (5.2.3.1)$$

Dabei gilt für die Reglerkonstanten:

$$\begin{aligned} h_0 &= k_3 * k_1 * k_2 & h_1 &= k_3 * (k_1 * k_2 + k_i) \\ h_2 &= k_3 * k_2 & h_3 &= k_3 \\ h_i &= k_3 * k_1 * k_i \end{aligned} \quad (5.2.3.2)$$

Das ausgehend vom kaskadierten Regler hergeleitete Reglergesetz (5.2.3.1) ist identisch mit dem Reglergesetz (5.2.2.3) des Zustandsreglers.

Die Reglerkonstanten des entstandenen Zustandsreglers erfüllen die Beziehung:

$$(h_1 - h_0) * h_0 = h_i * h_2 \quad (5.2.3.3)$$

Aus den Konstanten eines Zustandsreglers, der die Bedingung (5.2.3.3) erfüllt, berechnen sich die Konstanten des äquivalenten kaskadierten Reglers wie folgt:

$$\begin{aligned} k_1 &= h_0 / h_2 & k_2 &= h_2 / h_3 \\ k_3 &= h_3 & k_i &= (h_i * h_2) / (h_3 * h_0) \end{aligned} \quad (5.2.3.4)$$

Es soll nun versucht werden, das Reglergesetz des Zustandsregler gemäß (5.2.2.4) in die Kaskadenreglerstruktur zu überführen.

Mit dem Sollwert:

$$w_{soll} = h_1 / (h_1 - h_0) * (\varepsilon_{soll} - \varepsilon) \quad (5.2.3.5)$$

folgt aus (5.2.2.4) unter Benutzung von (5.1.3) die Gleichung:

$$\begin{aligned} u_{q,soll}^* &= h_0 * (h_1 - h_0) / h_1 * w_{soll} - h_2 * w - h_3 * i_q + \\ &+ (h_1 - h_0) * w_{soll} \, dt - (h_1 - h_0) * w \, dt \end{aligned}$$

Durch Umformung entsteht daraus:

$$\begin{aligned} u_{q,soll}^* &= h_0 * (h_1 - h_0) / h_1 * (w_{soll} - w) + (h_0 * (h_1 - h_0) / h_1 - h_2) * w - \\ &- h_3 * i_q + (h_1 - h_0) * (w_{soll} - w) \, dt \end{aligned}$$

Mit dem Sollwert:

$$\begin{aligned} i_{q,soll} &= h_0 * (h_1 - h_0) / (h_1 * h_3) * (w_{soll} - w) + (h_1 - h_0) / h_3 * (w_{soll} - w) \, dt + \\ &+ (h_0 * (h_1 - h_0) / h_1 - h_2) / h_3 * w \end{aligned} \quad (5.2.3.6)$$

folgt nun:

$$u_{q,soll}^* = h_3 * (i_{q,soll} - i_q) \quad (5.2.3.7)$$

Die Gleichungen (5.2.3.5) bis (5.2.3.7) bilden das Reglergesetz des zugehörigen kaskadierten Reglers.

Der im Reglergesetz (5.2.3.6) des Drehzahlreglers enthaltene zusätzliche Term $(h_0 * (h_1 - h_0) / h_1 - h_2) / h_3 * w$ kann als Kompensation eines drehzahlproportionalen Reibmoments aufgefasst werden.

Er verschwindet genau dann, wenn die Bedingung (5.2.3.3) erfüllt ist.

Für die Konstanten des entstandenen kaskadierten Reglers gilt:

$$\begin{aligned} k_1 &= h_1 / (h_1 - h_0) & k_2 &= h_0 * (h_1 - h_0) / (h_1 * h_3) \\ k_3 &= h_3 & k_1 &= (h_1 - h_0) / h_3 \\ k_4 &= (h_0 * (h_1 - h_0) / h_1 - h_2) / h_3 \end{aligned} \quad (5.2.3.8)$$

Somit ist gezeigt, daß die Zustandsreglerstruktur (5.2.2.4) äquivalent ist zu einer Kaskadenreglerstruktur, bei der der Drehzahlregler einen zusätzlichen drehzahlproportionalen Term liefert. Dabei ist $k_4 > 0$, wenn $h_0 * (h_1 - h_0) - h_2 * h_1 > 0$ ist.

Cevik /11/ verwendet einen Zustandsregler, bei dem $h_0 = 0$ ist. Der Sollwert ε_{soll} wirkt dann nur über den Integralteil auf den Regler. In diesem Fall gilt für die Konstanten des kaskadierten Reglers:

$$\begin{aligned} k_1 &= h_1/h_1 & k_2 &= 0 \\ k_3 &= h_3 & k_4 &= h_1/h_3 \\ k_4 &= -h_2/h_3 \end{aligned} \quad (5.2.3.9)$$

Ist $h_0 = h_1$, so kann die obige Umformung des Zustandsreglers in einen kaskadierten Regler nicht durchgeführt werden. Das Reglergesetz des Zustandsreglers lautet dann:

$$u^*_{q,soll} = h_1 * (\varepsilon_{soll} - \varepsilon) - h_2 * w - h_3 * i_q + h_1 * (\varepsilon_{soll} - \varepsilon) dt \quad (5.2.3.10)$$

Mit dem Sollwert:

$$w_{soll} = h_1/h_2 * (\varepsilon_{soll} - \varepsilon) + h_1/h_2 * (\varepsilon_{soll} - \varepsilon) dt \quad (5.2.3.11)$$

folgt aus (5.2.3.10) die Gleichung:

$$u^*_{q,soll} = h_2 * w_{soll} - h_2 * w - h_3 * i_q$$

$$u^*_{q,soll} = h_2 * (w_{soll} - w) - h_3 * i_q$$

Mit dem Sollwert:

$$i_{q,soll} = h_2/h_3 * (w_{soll} - w) \quad (5.2.3.12)$$

folgt nun:

$$u^*_{q,soll} = h_3 * (i_{q,soll} - i_q) \quad (5.2.3.13)$$

Es entsteht im Fall $h_0 = h_1$ also ein kaskadierter Regler mit einem PI-Lageregler und unterlagertem P-Drehzahl und P-Stromregler. Es wird deutlich, daß dieser Regler - wie schon beschrieben - überschwingen muß. Die Überschwingweite ist dabei von der Sprunghöhe abhängig.

Im folgenden sollen einige Eigenschaften der betrachteten Reglerstrukturen einander gegenübergestellt werden:

Zustandsregler:

1. Da hier kein Stromsollwert $i_{s,11}$ auftritt, fehlt die Möglichkeit zur Begrenzung des Iststroms über die die Begrenzung des Sollstroms. Zur Begrenzung des Iststroms ist hier eine Sonderfunktion erforderlich, die $u_{s,11} = 0$ setzt, solange $|i| > i_{max}$ ist.

Hierdurch kommt jedoch ein nichtlineares Element in das Reglergesetz, das das Verhalten des Systems in sehr unüberschaubarer Weise beeinflusst.

2. Zur Berechnung der Stellgröße $u_{s,11}$ sind 8 Multiplikationen erforderlich.

3. Die Berechnung der Stellgröße im Festkommaformat bereitet hier Schwierigkeiten, da zunächst die Multiplikationen und anschließend die Additionen und Subtraktionen ausgeführt werden müssen.

Probleme bereitet so beim gegebenen System bereits die Bestimmung der Terme $h_0 * z_{s,11}$ und $h_1 * z$ in Gleichung (5.2.2.3), da die Lagewerte 32-Bit-Werte sind, während Multiplikationen beim verwendeten Prozessor nur mit 16-Bit-Werten durchgeführt werden können. Dies führt zu zusätzlichem Rechenaufwand.

Kaskadenregler:

1. Das Auftreten des sollwertproportionalen Werts $i_{s,11}$ liefert eine Möglichkeit zur Begrenzung.

2. Zur Berechnung der Stellgröße $u_{s,11}$ sind 10 Multiplikationen erforderlich.

3. Die Tatsache, daß hier bei der Berechnung abwechselnd Differenzbildungen und Multiplikationen erfolgen, erweist sich als günstig bei Berechnungen im Festkommaformat.

Die bei Rechnungen im Festkommaformat nach jeder Multiplikation notwendige Begrenzerfunktion liefert einen möglicherweise begrenzten, jedoch vorzeichenrichtigen Wert, der in den nächsten Rechenschritt eingeht.

5.2.4 Kaskadierter Zustandsregler

Hierunter wird ein Lagezustandsregler mit unterlagertem Stromzustandsregler verstanden.

Das Reglergesetz dieses Reglers kann aus dem Reglergesetz des Zustandsreglers oder dem Reglergesetz des Kaskadenreglers hergeleitet werden. Hier wird es aus dem Reglergesetz des Zustandsreglers entwickelt.

Das Reglergesetz des Stromzustandsreglers nach (4.3.3) lautet:

$$u_{d,soll}^* = -k_2 \cdot i_d \quad (5.2.4.1)$$

$$u_{q,soll}^* = k_1 \cdot i_{q,soll} - k_2 \cdot i_q \quad (5.2.4.2)$$

$$\underline{u}_{soll} = \underline{u}_{soll}^* + j\omega \cdot \underline{i} + j\omega \quad (5.2.4.3)$$

Aus dem Reglergesetz (5.2.2.4) des Zustandsreglers folgt durch Gleichsetzen mit (5.2.4.2) :

$$\begin{aligned} k_1 \cdot i_{q,soll} - k_2 \cdot i_q &= h_1 \cdot (\varepsilon_{soll} - \varepsilon) + (h_0 - h_1) \cdot (\varepsilon_{soll} - \varepsilon_0) - \\ &- h_2 \cdot w - h_3 \cdot i_q + h_1 \cdot (\varepsilon_{soll} - \varepsilon) \, dt \end{aligned}$$

$$\begin{aligned} i_{q,soll} &= h_1 / k_1 \cdot (\varepsilon_{soll} - \varepsilon) + (h_0 - h_1) / k_1 \cdot (\varepsilon_{soll} - \varepsilon_0) - \\ &- h_2 / k_1 \cdot w - (h_3 - k_2) / k_1 \cdot i_q + h_1 / k_1 \cdot (\varepsilon_{soll} - \varepsilon) \, dt \end{aligned} \quad (5.2.4.4)$$

Damit ist das Reglergesetz des kaskadierten Zustandsreglers hergeleitet worden.

Das Reglergesetz vereinfacht sich, wenn $h_3 = k_2$ ist.

Der kaskadierte Zustandsregler besitzt gegenüber dem einfachen Zustandsregler den Vorteil, daß der hier auftretende Stromsollwert i_{soll} eine Möglichkeit zur Begrenzung des Stroms liefert.

Bild 5.3 zeigt die Struktur des kaskadierten Zustandsreglers.

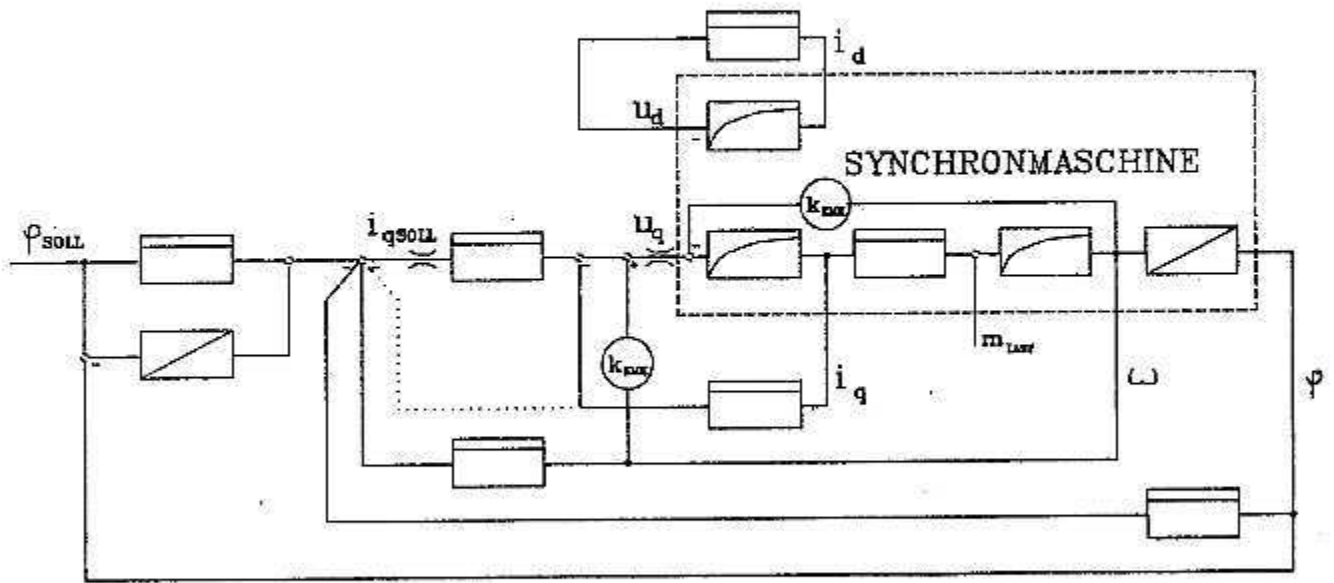


Bild 5.3 Strukturbild des kaskadierten Lage-Zustandsreglers

5.3 REGLERSTRUKTUREN MIT BERÜCKSICHTIGUNG DER TOTZEIT

5.3.1 ALLGEMEINES

Während im Abschnitt 5.2 eine zeitkontinuierliche Darstellung möglich war, ist hier wegen des rekursiven Charakters der Reglergesetze die zeitdiskrete Darstellung erforderlich. Die in Abschnitt 4.5.1 getroffenen Vereinbarungen bezüglich der Indizierung der zeitdiskreten Werte gelten auch hier.

5.3.2 BEOBACHTERFUNKTIONEN

Bei einer Regelung, bei der die Totzeit gleich der Abtastzeit ist, tritt das Problem auf, daß bei der Regelung auf den Zeitpunkt $n+2$ hin für die Bestimmung der Stellgröße $u^{*}_{s.o.l.l.,n+2}$ nur die Meßwerte zum Zeitpunkt n : i_n , w_n und ε_n zur Verfügung stehen.

Die Werte i_{n+1} , w_{n+1} und ε_{n+1} lassen sich näherungsweise durch Beobachterfunktionen bestimmen.

Die von den Beobachterfunktionen gelieferten Werte werden durch ein hochgestelltes B von den realen Werte zum Zeitpunkt $n+1$ unterschieden.

Damit erhält man:

$$i^B_{n+1} = a \cdot i_n + (1-a) \cdot u^{*}_{s.o.l.l.,n} \quad (5.3.2.1)$$

$$w^B_{n+1} = b_1 \cdot w_n + b_2 \cdot i_{q,n} + b_3 \cdot i^B_{q,n+1} \quad (5.3.2.2)$$

$$\varepsilon^B_{n+1} = \varepsilon_n + c_1 \cdot w_n + c_2 \cdot w^B_{n+1} \quad (5.3.2.3)$$

Die Konstanten der Beobachterfunktionen können direkt dem zeitdiskreten Übergangsmodell bestimmt werden. Aus regelungstechnischen Gründen können jedoch auch andere Konstanten gewählt werden.

Die durch die Funktionen (5.3.2.2) und (5.3.2.3) geschätzten Werte w_{n+1} und ε_{n+1} sind den vielfältigen Störgrößen des mechanischen Teils des Systems ausgesetzt. Die Schätzwerte w^B_{n+1} und

ε_{n+1}^B sind also unsicher. Dagegen ist der Wert i_{n+1} kaum Störgrößen ausgesetzt. Der Schätzwert i_{n+1}^B kann also als sicher gelten.

Werden die von den Beobachterfunktionen gelieferten Schätzwerte bei der Zustandsregelung verwendet, so führt dies zu einem Zustandsregler-Beobachter-System. Durch Einsetzen der Beobachterfunktionen kann dieses System wieder auf einen gewöhnlichen Zustandsregler zurückgeführt werden.

Das Arbeiten mit einem Zustandsregler-Beobachter-System hat insofern Vorteile, als die von den Beobachtern gelieferten Werte auch für die Transformationen und die Kompensationsrechnungen benutzt werden können. Allerdings steigt bei Verwendung eines Zustandsregler-Beobachter-Systems der Rechenaufwand stark an. Da dieser Rechenaufwand von dem betrachteten System in der vorgesehenen Taktzeit nicht zu leisten war, wurde dieser Ansatz nicht weiter verfolgt.

5.3.3 ZUSTANDSREGLER MIT TOTZEITBERÜCKSICHTIGUNG

Als zusätzliche Eingangsgröße des Reglers tritt hier die Stellgröße aus dem letzten Regelintervall auf.

Das Reglergesetz des Zustandsreglers mit Totzeitberücksichtigung lautet somit:

$$I_n = I_{n-1} + \varepsilon_{soll,n} - \varepsilon_n$$

$$u_{d,soll,n+1}^* = -h_3 \cdot i_{d,n} - h_4 \cdot u_{d,soll,n}^* \quad (5.3.3.1)$$

$$\begin{aligned} u_{q,soll,n+1}^* = & h_0 \cdot \varepsilon_{soll,n} - h_1 \cdot \varepsilon_n + (h_1 - h_0) \cdot \varepsilon_0 - h_2 \cdot w_n - \\ & - h_3 \cdot i_{q,n} - h_4 \cdot u_{q,soll,n}^* + h_1 \cdot I_n \end{aligned} \quad (5.3.3.2)$$

Wie bereits in Abschnitt 5.2.2 gezeigt wurde, kann der Rechenaufwand verringert werden, wenn die Kompensation der Polradspannung in das Zustandsreglergesetz aufgenommen wird.

5.3.4 KASKADIERTER ZUSTANDSREGLER MIT TOTZEITBERÜCKSICHTIGUNG

Das Reglergesetz dieses Reglers wird es aus dem Reglergesetz des Zustandsreglers (5.3.3.2) durch Gleichsetzen des Reglergesetzes mit dem Gesetz des Stromzustandsregler mit Totzeitberücksichtigung nach Abschnitt 4.5.2 hergeleitet. Dieser hat das Reglergesetz:

$$\underline{u}^*_{s011,n+1} = k_1 * \underline{i}_{s011,n} - k_2 * \underline{i}_n - k_3 * \underline{u}^*_{s011,n} \quad (5.3.4.1)$$

Hieraus folgt mit (5.3.3.2) :

$$\begin{aligned} i_{q,s011,n} = & h_0/k_1 * \varepsilon_{s011,n} - h_1/k_1 * \varepsilon_n + (h_1 - h_0)/h_1 * \varepsilon_0 - h_2/k_1 * w_n - \\ & - (h_3 - k_2)/k_1 * i_{q,n} - (h_4 - k_3)/k_1 * u^*_{q,s011,n} + h_1/k_1 * I_n \end{aligned} \quad (5.3.4.2)$$

Das Reglergesetz vereinfacht sich, wenn $h_4 = k_3$ und/oder $h_3 = k_2$ ist.

Anstelle des Stromzustandsreglers im rotorfesten System kann hier auch der in Abschnitt 4.5.4 beschriebene Stromzustandsreglers im statorfesten System als unterlagerter Stromregler verwendet werden.

5.3.5 KASKADENREGLER MIT TOTZEITBERÜCKSICHTIGUNG

Ausgegangen wird von Gleichung (5.3.4.2) . Diese wird in Kaskadenform umgerechnet. Zur Vereinfachung werden in (5.3.4.2) neue Konstanten eingeführt:

$$\begin{aligned} i_{q,s011,n} = & g_0 * \varepsilon_{s011,n} - g_1 * \varepsilon_n + (g_1 - g_0) * \varepsilon_0 - g_2 * w_n - \\ & - g_3 * i_{q,n} - g_4 * u^*_{q,s011,n} + g_1 * \sum_{k=0}^n (\varepsilon_{s011,k} - \varepsilon_k) \end{aligned} \quad (5.3.5.1)$$

Mit:

$$w_{s011,n} = g_1 / (g_1 - g_0) * (\varepsilon_{s011,n} - \varepsilon_n) \quad (5.3.5.2)$$

folgt:

$$i_{q, \text{ Soll, n}} =$$

$$= g_0 * (g_1 - g_0) / g_1 * w_{\text{ Soll, n}} - g_2 * w_n - g_3 * i_{q, n} - g_4 * u^*_{q, \text{ Soll, n}} +$$

$$+ (g_1 - g_0) * \sum_0^n w_{\text{ Soll, k}} - (g_1 - g_0) * \sum_0^n w_k$$

$$i_{q, \text{ Soll, n}} =$$

$$g_0 * (g_1 - g_0) / g_1 * (w_{\text{ Soll, n}} - w_n) - g_3 * i_{q, n} - g_4 * u^*_{q, \text{ Soll, n}} +$$

$$+ (g_1 - g_0) * \sum_0^n (w_{\text{ Soll, k}} - w_k) + (g_0 * (g_1 - g_0) / g_1 - g_2) * w_n$$

(5.3.5.3)

Dieses Reglergesetz des Drehzahlreglers unterscheidet sich von dem eines PI-Reglers durch das Hinzutreten des Ausdrucks:

$$(g_0 * (g_1 - g_0) / g_1 - g_2) * w_n - g_3 * i_{q, n} - g_4 * u^*_{q, \text{ Soll, n}}$$

Der erste Summand stellt wieder eine Kompensation des Reibmoments dar. Die Konstante g_4 kann durch Dimensionierung des Stromreglers zu Null gemacht werden. Der verbleibende Term $- g_3 * i_{q, n}$ kann gedeutet werden als zusätzliche Störgößenkompensation.

6. ENTWURF DER REGLER

6.1 ÜBERTRAGUNGSFUNKTIONEN DER REGELSTRECKE

Ausgegangen wird von dem in Kapitel 2 abgeleiteten zeitdiskreten Zustandsmodell der PM-Synchronmaschine nach Abschnitt 2.2. Verwendet wird die in Abschnitt 3.7.7 beschriebene Methode des Zustandsreglerentwurf bei einer Strecke mit Blockstruktur. Entworfen wird ein kaskadierter Zustandsregler nach Abschnitt 5.2.4. Das Reglergesetz dieses Regler kann dann sowohl in die Kaskadenreglerstruktur als auch in die reine Zustandsreglerstruktur umgerechnet werden.

Mit dem Zustandsvektor:

$$\mathbf{x} = [i_q, w, \varepsilon]^T \quad (6.1.1)$$

liefert die Anwendung der Z-Transformation auf das zeitdiskrete Zustandsmodell der PM-Synchronmaschine:

$$z \cdot \mathbf{x} = \begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & 1 \end{bmatrix} \cdot \mathbf{x} + \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \cdot u^*_q \quad (6.1.2)$$

Die hierin auftretenden Konstanten sind durch die Gleichungen (2.2.15) und (2.2.16) gegeben.

Bei Berücksichtigung der Totzeit entsteht mit dem Zustandsvektor:

$$\mathbf{x}_T = [i_q, w, \varepsilon, u^*_q]^T \quad (6.1.3)$$

das erweiterte Modell:

$$z^*x = \begin{bmatrix} a_{11} & 0 & 0 & h_1 \\ a_{21} & a_{22} & 0 & h_2 \\ a_{31} & a_{32} & 1 & h_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} *x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} * u^*_q \quad (6.1.4)$$

Die Totzeit ist hier dem Steckeneingang zugeordnet worden.

Um den Zustandsreglerentwurf bei einer Strecke mit Blockstruktur anwenden zu können, ist es notwendig, die Übertragungsfunktionen $G_1(z)$, $G_2(z)$ und $G_3(z)$ der Blöcke zu berechnen.

Aus (6.1.4) folgt:

$$G_1(z) = \frac{i_q(z)}{u^*_q(z)} = \frac{h_1}{z^*(z-a_{11})} \quad (6.1.5)$$

Die ÜF des elektrischen Teils der Strecke ohne Berücksichtigung der Totzeit lautet:

$$G_0(z) = \frac{i_q(z)}{u^*_q(z)} = \frac{h_1}{z-a_{11}} \quad (6.1.6)$$

Mit der zweiten Zeile von (6.1.4)

$$(z-a_{22})*w = a_{21}*i_q + h_2*u^*_q \quad (6.1.7)$$

erhält man:

$$(z-a_{22})*w = (a_{21} + (z-a_{11})*h_2/h_1)*i_q$$

Somit gilt:

$$G_2(z) = \frac{w(z)}{i_q(z)} = \frac{a_{21} + (z-a_{11})*h_2/h_1}{z-a_{22}} \quad (6.1.8)$$

$$G_2(z) = \frac{A*z + B}{z - a_{22}}$$

mit: $A = h_2/h_1$ und $B = a_{21} - a_{11}*h_2/h_1$

Aus der dritten Zeile von (6.1.4):

$$(z-1)*\varepsilon = a_{31}*i_q + a_{32}*w + h_3*u^{**}_q$$

erhält man mit (6.1.6):

$$(z-1)*\varepsilon = (a_{31}+h_3/G_0)*i_q + a_{32}*w$$

Mit (6.1.8) erhält man:

$$(z-1)*\varepsilon = ((a_{31}+h_3/G_0)/G_2 + a_{32})*w$$

$$(z-1)*\varepsilon = \frac{(a_{31}*Z_0+h_3*N_0)*N_2 + Z_0*Z_2*a_{32}}{Z_0*Z_2} * w$$

$$G_3(z) = \frac{\varepsilon(z)}{w(z)} = \frac{(a_{31}h_1+h_3*(z-a_{11}))* (z-a_{22})+(h_1a_{21}+(z-a_{11})*h_2)a_{32}}{(z-1)*(h_1a_{21}+(z-a_{11})*h_2)}$$

$$G_3(z) = \frac{h_3*z^2 - (a_{22}h_3+a_{32}h_2+a_{11}h_3+a_{31}h_1)*z - (a_{31}a_{22}h_1-a_{21}a_{32}h_1)}{h_2*z^2 + (a_{21}h_1-a_{11}h_2-h_2)*z - (a_{21}h_1-a_{11}h_2)} \quad (6.1.9)$$

Geht man von dem Übergangsmodell (2.2.20) für den mechanischen Teil aus und setzt für die Eingangsgröße i_q den Näherungswert:

$$i_q = (i_{q,n+1} + i_{q,n})/2 \quad (6.1.10)$$

ein, so folgt mit den in Abschnitt 2.2 verwendeten Abkürzungen:

$$(z-a_{22})*w(z) = (1-e^{-CT})*a*(z+1)/2 * i_q(z) \quad (6.1.11)$$

und

$$(z-1)*\varepsilon = a_{32}*w + (T + 1/C*(e^{-CT}-1))*a*(z+1)/2 * i_q(z) \quad (6.1.12)$$

Es werden nun die Abkürzungen

$$D = (1-e^{-CT})*a/2 \quad E = (T + 1/C*(e^{-CT}-1))*a/2$$

eingeführt.

Aus (6.1.11) erhält man die ÜF:

$$G_2(z) = \frac{w(z)}{i_q(z)} = \frac{D*(z+1)}{z-a_{22}} \quad (6.1.13)$$

Aus (6.1.12) folgt:

$$(z-1)*\varepsilon = a_{32} * w + E*(z+1)*i_q(z)$$

$$(z-1)*\varepsilon = a_{32} * w + E*(z+1)*N_2/Z_2 * w$$

Hieraus erhält man die ÜF:

$$G_3(z) = \frac{E/D*z + a_{32} - E/D*a_{22}}{z-1}$$

$$G_3(z) = \frac{G*z+H}{z-1} \quad (6.1.14)$$

Dabei gilt:

$$G = E/D = (T + 1/C*(e^{-CT}-1))/(e^{-CT}-1) = T/(e^{-CT}-1) + 1/C \quad (6.1.15)$$

und

$$H = a_{32} - G*a_{22} = 1/C*(e^{-CT}-1) - G*e^{-CT}$$

$$= 1/C*(e^{-CT}-1) - (T/(e^{-CT}-1) + 1/C)*e^{-CT}$$

$$= -1/C - T*e^{-CT}/(e^{-CT}-1) \quad (6.1.16)$$

Der Vorteil dieser nicht ganz exakten Berechnung liegt darin, daß eine einfachere Funktion $G_3(z)$ entsteht. Die gemachte Vereinfachung ist aber insoweit zulässig, als das Übergangsverhalten des Stroms ohnehin nur näherungsweise beschreibbar ist. Beim nachfolgend beschriebenen Reglerentwurf wird von den ÜF (6.1.5), (6.1.13) und (6.1.14) ausgegangen.

6.2 ENTWURF DES STROMREGLERS

Als Stromregler wird ein Regler mit Totzeitberücksichtigung nach Abschnitt 4.5 verwendet. Aus den Gleichungen (4.5.2.4) und (4.5.4.5) folgt, daß die Stromregelstrecke bei jeweils geeignet definiertem u^* und Vernachlässigung von Abweichungen, die bei

der Behandlung der Stromregler beschrieben wurden, im statorfesten und im rotorfesten System ein identisches Übergangsverhalten zeigt:

$$\underline{i}_{n+1} = e^{-T} * \underline{i}_n + (1 - e^{-T}) * \underline{u}_n^* \quad (6.2.1)$$

Auch die sich bei Berücksichtigung der Totzeit ergebenden Reglergesetze (4.5.2.6) und (4.5.4.9) besitzen die gleiche Struktur:

$$\underline{u}_{soll, n+1}^* = k_1 * \underline{i}_{soll, n} - k_2 * e^{-T} * \underline{i}_n - k_2 * (1 - e^{-T}) * \underline{u}_n^* \quad (6.2.2)$$

Mit $a = e^{-T}$ folgt durch Anwendung der Z-Transformation:

$$z * \underline{i} = a * \underline{i} + (1 - a) * \underline{u}^* \quad (6.2.3)$$

und

$$z * \underline{u}_{soll}^* = k_1 * \underline{i}_{soll} - k_2 * a * \underline{i} - k_2 * (1 - a) * \underline{u}^* \quad (6.2.4)$$

Indem nun $\underline{u}_{soll}^* = \underline{u}^*$ gesetzt wird, erhält man:

$$(z + k_2 * (1 - a)) * \underline{u}^* = k_1 * \underline{i}_{soll} - k_2 * a * \underline{i}$$

$$(z - a) * \underline{i} = (1 - a) * (k_1 * \underline{i}_{soll} - k_2 * a * \underline{i}) / (z + k_2 * (1 - a))$$

$$(z - a + (1 - a) * k_2 * a) / (z + k_2 * (1 - a)) * \underline{i} =$$

$$= (1 - a) * k_1 * \underline{i}_{soll} / (z + k_2 * (1 - a))$$

Die Z-ÜF des geschlossenen Regelkreises lautet somit:

$$G_F(z) = \frac{(1 - a) * k_1}{z^2 + (k_2 * (1 - a) - a) * z} \quad (6.2.5)$$

Damit hieraus die ÜF $G_F(z) = 1/z^2$ des DEAD-BEAT-Reglers entsteht, muß gelten:

$$k_1 = 1 / (1 - e^{-T}) \quad \text{und} \quad k_2 = e^{-T} / (1 - e^{-T})$$

Damit erhält man als Reglergesetz des DEAD-BEAT-Stromreglers:

$$\underline{u}_{soll, n+1}^* = 1 / (1 - e^{-T}) * \underline{i}_{soll, n} - e^{-2T} / (1 - e^{-T}) * \underline{i}_n - e^{-T} * \underline{u}_n^*$$

Um Stabilitätsprobleme zu vermeiden, kann es hier sinnvoll sein,

einen etwas weicher eingestellten Zustandsregler zu verwenden.

6.3 ENTWURF EINES DREHZAHLZUSTANDSREGLERS

Das Reglergesetz des überlagerten Drehzahlzustandsreglers mit I-Anteil lautet:

$$u_{qsoll,n} = k_0 * w_{soll,n} - k_I * w_n + k_I * I_n \quad (6.3.1)$$

Dabei gilt für den Integralanteil I_n :

$$I_n = I_{n-1} + w_{soll,n} - w_n \quad (6.3.2)$$

Da die Totzeit der Stromregelschleife zugerechnet wird, darf sie hier nicht berücksichtigt werden.

Im Z-Bereich folgt:

$$u_{qsoll}(z) = k_0 * w_{soll} - k_I * w + k_I * I \quad (6.3.3)$$

$$I(z) = z/(z-1) * (w_{soll} - w) \quad (6.3.4)$$

Daraus erhält man:

$$(z-1) * u_{qsoll}(z) = ((z-1) * k_0 + z * k_I) * w_{soll} - ((z-1) * k_I + z * k_I) * w \quad (6.3.5)$$

Mit der OF $i_q = 1/z^2 * u_q$ des Stromreglers folgt:

$$(z-1) * z^2 * i_q = ((z-1) * k_0 + z * k_I) * w_{soll} - ((z-1) * k_I + z * k_I) * w$$

Mit der OF (6.1.13) erhält man:

$$\begin{aligned} (z-1) * z^2 * (z-a_{22}) * w &= \\ &= D * (z+1) * ((z-1) * k_0 + z * k_I) * w_{soll} - ((z-1) * k_I + z * k_I) * w \\ ((z-1) * z^2 * (z-a_{22}) + D * (z+1) * ((k_I + k_I) * z - k_I)) * w &= \end{aligned}$$

$$= D \cdot (z+1) \cdot ((k_0 + k_1) \cdot z - k_0) \cdot W_{soll}$$

$$G_F(z) = \frac{D \cdot (z+1) \cdot ((k_0 + k_1) \cdot z - k_0)}{(z-1) \cdot z^2 \cdot (z-a_{22}) + D \cdot (z+1) \cdot ((k_1 + k_0) \cdot z - k_1)} \quad (6.3.6)$$

Die Polstellen des Nennerpolynoms lassen sich nun nicht mehr völlig frei wählen, da zur Bestimmung der 4 Polstellen nur die zwei Konstanten k_2 und k_3 zur Verfügung stehen.

Die bleibende Regelabweichung ist 0, da $G_F(1) = 1$ ist.

Für den Zähler $Z_E(z)$ der Fehler-ÜF gilt:

$$Z_E(z) = N_F - Z_F = (z-1) \cdot z^2 \cdot (z-a_{22}) + D \cdot (z+1) \cdot (k_1 - k_0) \cdot (z-1)$$

$$Z_E(z) = (z-1) \cdot (z^2 \cdot (z-a_{22}) + D \cdot (k_1 - k_0) \cdot (z+1)) \quad (6.3.7)$$

Damit $e_z = 0$ ist, muß nun gelten:

$$(1-a_{22}) + D \cdot (k_1 - k_0) \cdot 2 = 0$$

Daraus folgt:

$$k_0 = k_1 + (1-a_{22}) / (2 \cdot D) \quad (6.3.8)$$

Durch Einsetzen der Werte der Abkürzungen erhält man nun:

$$(1-a_{22}) / (2 \cdot D) = (1-e^{-cT}) / (2 \cdot (1-e^{-cT}) \cdot a/2) = 1/a = k_R / k_{MOM}$$

Eine Umformung des Reglergesetzes (6.3.1) liefert bei Gültigkeit von (6.3.8) :

$$u_{q_{soll,n}} = k_R / k_{MOM} \cdot W_{soll,n} + k_I \cdot (W_{soll,n} - W_0) + k_I \cdot I_n \quad (6.3.9)$$

Der Term $k_R / k_{MOM} \cdot W_{soll,n}$ im Reglergesetz bewirkt das Auftreten des q-Stromterms $k_R / k_{MOM} \cdot W_{soll,n}$. Dieser bewirkt das Moment $k_R \cdot W_{soll,n}$. Es entsteht also ein PI-Regler mit zusätzlicher Vorsteuerung zum Ausgleich des Reibmoments. Man sieht hier, daß es bei Verwendung der Z-Transformation möglich ist, auch einfache

Zusammenhänge auf kompliziertem Wege herzuleiten.

6.4 ENTWURF DES LAGEZUSTANDSREGLERS

Das Reglergesetz des überlagerten Lagezustandsreglers mit I-Anteil lautet:

$$u_{qsoll,n} = k_0 * \varepsilon_{soll,n} - k_1 * \varepsilon_n - k_2 * w_n + k_I * I_n \quad (6.4.1)$$

Dabei gilt für den Integralanteil I_n :

$$I_n = I_{n-1} + \varepsilon_{soll,n} - \varepsilon_n \quad (6.4.2)$$

Der zur Erzielung der Lageinvarianz eingeführte Term braucht hier nicht berücksichtigt zu werden, da $\varepsilon_0 = 0$ vorausgesetzt wird. Im Z-Bereich folgt:

$$\begin{aligned} (z-1) * u_{qsoll} &= \\ &= ((z-1) * k_0 + z * k_I) * \varepsilon_{soll} - ((z-1) * k_1 + z * k_I) * \varepsilon - (z-1) * k_2 * w \end{aligned}$$

Mit der ÜF des Stromreglers und der ÜF (6.1.13) erhält man:

$$\begin{aligned} (z-1) * z^2 * (z - a_{22}) * w &= \\ = D * (z+1) * (((k_0 + k_I) * z - k_0) * \varepsilon_{soll} - ((k_1 + k_I) * z - k_I) * \varepsilon - (z-1) * k_2 * w) \end{aligned}$$

$$\begin{aligned} (z-1) * (z^2 * (z - a_{22}) + D * (z+1) * k_2) * w &= \\ = D * (z+1) * (((k_0 + k_I) * z - k_0) * \varepsilon_{soll} - ((k_1 + k_I) * z - k_I) * \varepsilon) \end{aligned}$$

Mit der ÜF (6.1.14) erhält man:

$$\begin{aligned} (z-1)^2 * (z^2 * (z - a_{22}) + D * (z+1) * k_2) * \varepsilon &= \\ = (G * z + H) * D * (z+1) * (((k_0 + k_I) * z - k_0) * \varepsilon_{soll} - ((k_1 + k_I) * z - k_I) * \varepsilon) = \\ ((z-1)^2 * (z^2 * (z - a_{22}) + D * (z+1) * k_2) + \\ + (G * z + H) * D * (z+1) * ((k_1 + k_I) * z - k_I)) * \varepsilon &= \\ = (G * z + H) * D * (z+1) * ((k_0 + k_I) * z - k_0) * \varepsilon_{soll} \end{aligned}$$

Somit folgt:

$$G_F(z) =$$

$$\frac{(G \cdot z + H) \cdot D \cdot (z+1) \cdot ((k_0 + k_1) \cdot z - k_0)}{(z-1)^2 \cdot (z^2 \cdot (z - a_{22}) + D \cdot (z+1) \cdot k_3) + (G \cdot z + H) \cdot D \cdot (z+1) \cdot ((k_1 + k_1) \cdot z - k_1)} \quad (6.4.3)$$

Die Polstellen des Nennerpolynoms lassen sich auch hier nicht mehr völlig frei wählen, da zur Bestimmung der 5 Polstellen nur die drei Konstanten k_2, k_3 und k_4 zur Verfügung stehen. Die bleibende Regelabweichung ist 0.

Für den Zähler $Z_E(z)$ der Fehler-ÜF gilt:

$$\begin{aligned} Z_E(z) &= \\ &= (z-1)^2 \cdot (z^2 \cdot (z - a_{22}) + D \cdot (z+1) \cdot k_3) + (G \cdot z + H) \cdot D \cdot (z+1) \cdot (k_1 - k_0) \cdot (z-1) \\ Z_E(z) &= \\ &= (z-1) \cdot ((z-1) \cdot (z^2 \cdot (z - a_{22}) + D \cdot (z+1) \cdot k_3) + (G \cdot z + H) \cdot D \cdot (z+1) \cdot (k_1 - k_0)) \end{aligned}$$

Damit $e_2 = 0$ ist, muß nun gelten:

$$(G+H) \cdot D \cdot 2 \cdot (k_1 - k_0) = 0 \quad (6.4.4)$$

Aus $e_2 = 0$ folgt also $k_1 = k_0$. Da ein solcher Regler aber - wie bereits gezeigt - überschwingen muß, läßt sich $e_2 = 0$ nicht realisieren, wenn ein Überschwingen vermieden werden soll.

7. DIE SIMULATION DER REGELUNG

7.1 ZIELSTELLUNG

Die Realisierung des Simulationsprogramms erfolgte unter folgender Zielstellung:

- Simulation des Verhaltens der Synchronmaschine
- Einbeziehung des Spannungszeigersyntheseverfahrens
- Wählbare Maschinenparameter
- Wählbare Lastparameter
- Wählbare Anfangsbedingungen
- Einbeziehung der Eigenschaften der Meßgrößenaufnehmer
- Einbeziehung von stochastischen Modellfehlern
- Wählbare Reglertypen und Reglerparameter
- Anzeige der Daten auf dem Bildschirm in wählbarem Zeitabstand.
- Möglichkeit zum Speichern wählbarer Daten in wählbarem Zeitabstand in eine Wertedatei bei gleichzeitiger Erstellung einer Datei mit den zugehörigen Parametern
- Möglichkeit zum Erstellen und Lesen von Dateien, die alle einstellbaren Parameter des Simulationsprogramms enthalten.

7.2 PROGRAMMIERSPRACHE

Das Programm zur Simulation des Verhaltens der PM-Synchronmaschine wurde mit Turbo-Pascal (TP) in der Version TURBO-87 auf einem PC erstellt. Diese Programmversion benutzt den Coprozessor 80x87 des PC's. Hierdurch wird zum einen eine höhere Rechengeschwindigkeit erzielt und zum anderen eine höhere Genauigkeit der Rechnungen bewirkt, da bei reellen Berechnungen eine größere Stellenzahl verwendet wird.

Der TP-Editor kann Quelldateien bis zu einer Größe von ca 64 kB verwalten. Diese Einschränkung wird hier durch Verwendung von INCLUDE-Dateien umgangen.

Da der TP-Compiler ausführbare Dateien im COM-Format erzeugt, dürfen diese eine maximale Länge von 64 kB besitzen. Auch diese Beschränkung wird hier durch Benutzung des Overlay-Mechanismus

umgangen. Der Overlay-Mechanismus arbeitet so, daß Prozeduren, die mit dem vorangestellten Bezeichner " overlay " gekennzeichnet sind, gemeinsam in eine Overlay-Datei compiliert werden. In der COM-Datei wird ein Bereich freigehalten, der durch die Länge der längsten Overlay-Prozedur bestimmt wird. Wird während des Programmablaufs eine Overlay-Prozedur aufgerufen, so wird zunächst geprüft, ob sich diese bereits im Overlaybereich befindet. Ist das nicht der Fall, so wird sie aus der Overlay-Datei geladen. Unter Verwendung des INCLUDE-Befehls und des OVERLAY-Mechanismus lassen sich also nahezu beliebig große Programmpakete erzeugen. Auch bezüglich der Daten, die TP verwalten kann, besteht eine Begrenzung auf 64 kB. In dem hier beschriebenen Programm reichte dieser Bereich zur Aufnahme der Variablen aus. Die Beschränkung der Datenmenge kann durch Verwendung von Variablen, die durch Zeiger angesprochen werden, aufgehoben werden. Es läßt sich so der gesamte von DOS verwaltete Speicherbereich des Computers ausschöpfen.

7.3 GROBSTRUKTUR DES PROGRAMMS - MENUES

Bild 7.1 zeigt die Grobstruktur des Programms. Der Modul "Parameterinit" führt die Initialisierung der Simulationsparameter und der internen Variablen durch. Vor der Wahl eines Menues oder einer Funktion erfolgt durch den Modul "Umrechnung" die Umrechnung der absoluten Werte in bezogene Werte. Die im Hauptmenue wählbaren Punkte werden unterschieden in Menues und Funktionen. Folgende Menues sind wählbar:

- | | |
|--------------------|-------------------------|
| 1. Maschinendaten | 2. Lastdaten |
| 3. Anfangswerte | 4. Stromreglerparameter |
| 5. Reglerparameter | 6. Simulationsparameter |
| 7. Speichergrößen | |

Folgende Funktionen sind wählbar:

- | | |
|--|-----------------------------------|
| 8. Simulationsparameter lesen | 9. Simulationsparameter speichern |
| 10. Zustandsreglerkonstanten berechnen | |
| 11. Simulation durchführen | 12. Programmende |

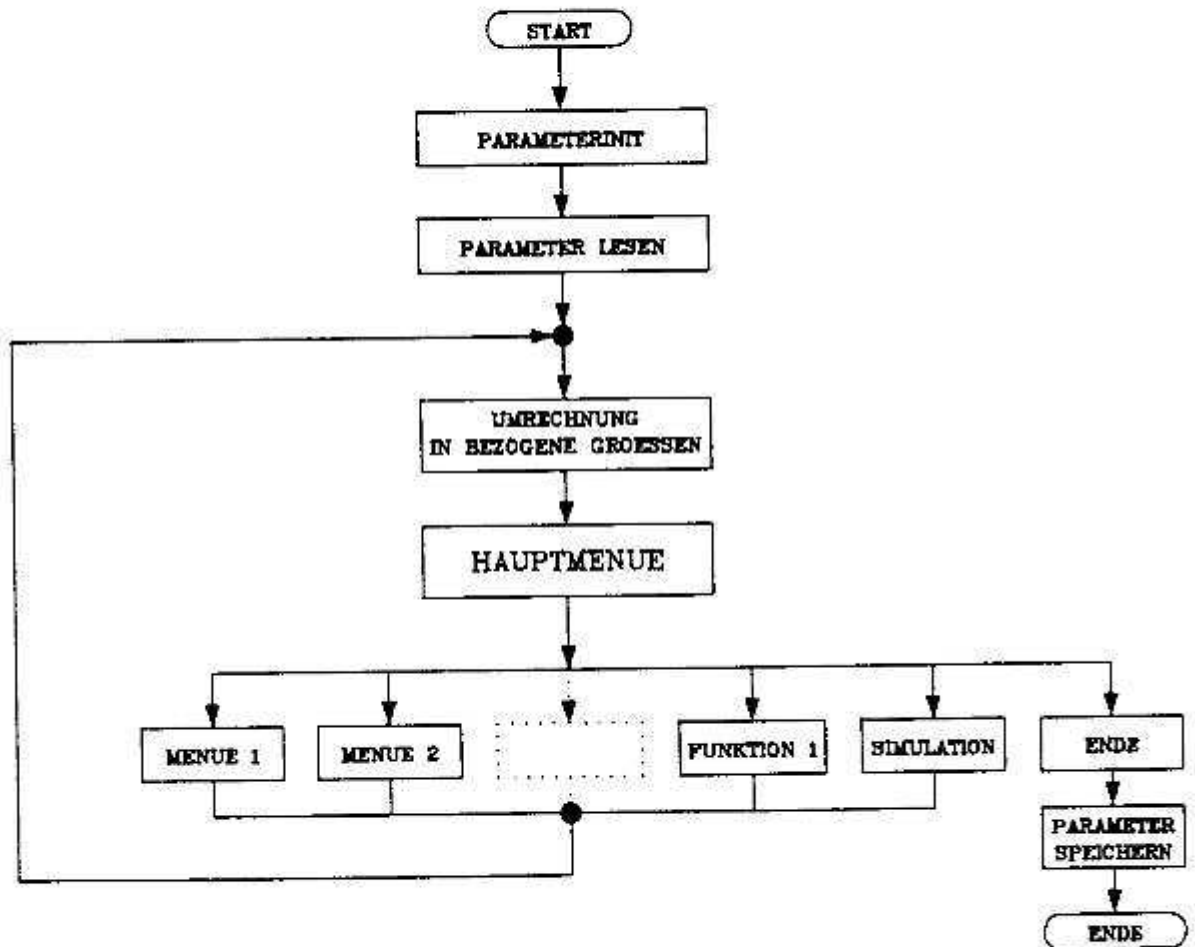


Bild 7.1 Vereinfachtes Flußdiagramm des Simulationsprogramms

7.4 BESCHREIBUNG DER MENUES UND FUNKTIONEN

7.4.1 Menue "Maschinendaten"

Bestimmt werden können hier Strangwiderstand, Stranginduktivität, EMK-Konstante, Rotorträgheitsmoment, Zwischenkreisspannung und maximal zulässiger Strom.

Bezüglich der Meßwertaufnehmer können folgende Parameter gewählt werden:

- Auflösung des Lagegebers
- maximaler prozentualer Strommeßfehler
- Art des Strommeßverfahrens (Momentanwert, Mittelwert)
- Auflösung bei der AD-Wandlung des Stroms
- Anzahl der Reglertakte für die Drehzahlbestimmung

Ferner kann ein maximaler prozentualer Systemfehler festgelegt werden. Dieser beschreibt stochastische Abweichungen des System vom zugrundeliegenden Modell. Realisiert wird dies, indem die Stromwerte am Ende jedes Pulsmodulationstakts gesteuert durch einen Zufallsgenerator verfälscht werden.

7.4.2 Menue "Lastdaten"

Hier können das Trägheitsmoment der Last, das Lastmoment und der Reibungskoeffizient festgelegt werden. Ferner kann hier eine Zeitfunktion für das Lastmoment festgelegt werden. Wählbar sind folgende Funktionen:

- | | |
|--------------------|------------------|
| - Sprungfunktion | - Rampenfunktion |
| - Rechteckfunktion | - Sinusfunktion |

7.4.3 Menue "Anfangswerte"

Bestimmt werden können hier Rotorwinkel, Rotordrehzahl, Betrag und Phase des Statorstroms.

7.4.4 Menue "Stromreglerparameter"

Hier sind folgende Menuepunkte wählbar:

- Wahl des Stromreglers

Bei der Wahl eines Stromreglers erscheint der zu diesem Regler gehörende Konstantensatz auf dem Bildschirm. Dieser besteht aus fünf Konstanten, die geändert werden können. Die wählbaren Stromregler werden später beschrieben.

- Bestimmung Pulsmodulationstaktzeit des 3-Vektor-Verfahrens

- Festlegung der Reglertaktzeit als ganzzahliges Vielfaches der Pulsmodulationstaktzeit

- Wahl der Vorsteuerungsart

Verfügbare Vorsteuerungsarten sind:

- a) Keine Vorsteuerung
- b) Vorsteuerung nach Orlik
- c) verlustminimale Vorsteuerung

- Wahl der Spannungsbegrenzungsart

Wählbar sind folgende Spannungsbegrenzungsarten:

- a) proportionale Spannungsbegrenzung
- b) Spannungsbegrenzung mit d-Priorität
- c) Spannungsbegrenzung mit q-Priorität
- d) Komplexe Spannungsbegrenzung

- Wahl der Kompensationsart

Gewählt werden kann hier, ob bei den in den Reglergesetzen auftretenden Kompensationstermen $i_{k,mp}$ die Stromistwerte oder Stromsollwerte verwendet werden sollen.

- Spannungszeigernachführung

Hier kann bestimmt werden, ob der Spannungszeiger bei jedem PWM-Takt entsprechend der Drehgeschwindigkeit des Rotors nachgeführt werden soll.

7.4.5 Menue "Reglerparameter"

In diesem Menue erfolgt die Wahl des übergeordneten Reglers (Lage- bzw. Drehzahlregler).

Die hier wählbare Regler arbeiten stets mit unterlagertem Stromregler. Ausgangsgröße der Regler ist das Sollmoment m_{soll} . Diesem Sollmoment ist der Sollstrom $i_{a,soll}$ proportional.

Bei Wahl eines Reglers erscheint der zu diesem Regler gehörende Konstantensatz auf dem Bildschirm. Dieser besteht aus jeweils fünf Konstanten, die geändert werden können.

Weiter erfolgt in diesem Menue die Festlegung der Sollwerte. Bestimmt werden kann hier:

- die Solldrehzahl
- die Sollage in der Form Sollwinkel und Sollumdrehungen
- das Sollmoment.

Für diese Sollwerte kann neben einem konstanten Wert jeweils eine Zeitfunktion vorgeben werden. Es sind hier die gleichen Zeitfunktionen wählbar wie für das Lastmoment. Die gewählten Sollwerte wirken jeweils nur abhängig von der gewählten Reglerart.

7.4.6 Menue "Speichergrößen"

Folgende Größen können gespeichert werden:

- | | |
|--------------------------------|--------------------------------|
| - Rotorwinkel | - Rotordrehzahl |
| - Rotorbeschleunigung | - Rotorsollwinkel |
| - Rotorsolldrehzahl | - Statorstrombetrag i |
| - Statorstromwinkel | - Strom i_x |
| - Strom i_y | - Zwischenkreisstrom i_{zx} |
| - Strom i_d | - Strom i_q |
| - Spannungssollwert $u_d.soll$ | - Spannungssollwert $u_q.soll$ |
| - Stromsollwert $i_d.soll$ | - Stromsollwert $i_q.soll$ |
| - Spannungsiswert u_d | - Spannungsiswert u_q |

7.4.7 Menue "Simulationsparameter"

Bestimmt werden kann hier die Anzahl der Schritte des Runge-Kutta-Verfahrens bezogen auf die elektrische Zeitkonstante T_e der Maschine. Als ausreichend erwies sich hier eine Schrittzahl von 250.

Ferner kann festgelegt werden, ob bei der Bildschirmausgabe und bei der Speicherung der Werte die bezogenen oder die absoluten Werte verwendet werden sollen.

Bestimmt werden kann auch, ob nur die Werte am Ende der PWM-Takte oder auch Zwischenwerte während der PWM-Takte gezeigt und/oder

gespeichert werden sollen. Zusätzlich kann hier eine Kontrollberechnung eingeschaltet werden, die nicht mit dem Runge-Kutta-Verfahren arbeitet.

Gewählt werden kann außerdem, ob die Simulation mit einer Reglertotzeit gleich der Reglertaktzeit oder ohne Reglertotzeit durchgeführt werden soll.

7.4.8 Funktion "Simulationsparameter lesen"

Es wird eine wählbare Parameterdatei gelesen. Die Parameterdateien enthalten alle wesentlichen einstellbaren Parameter des Programms. Die Namen der Parameterdateien besitzen einheitlich das Suffix "SPD", das bei der Wahl nicht eingegeben zu werden braucht.

Existiert die angegebene Parameterdatei nicht, so wird auf dem Bildschirm eine Liste aller vorhandenen Parameterdateien angezeigt und anschließend erneut die Aufforderung zur Eingabe des Dateinamens ausgegeben.

7.4.9 Funktion "Simulationsparameter speichern"

Die gegenwärtigen Parameter des Programms werden in eine wählbare Parameterdatei geschrieben. Wurde bereits eine Parameterdatei gelesen, so wird deren Name als Default vorgegeben.

7.4.10 Funktion "Zustandsreglerkonstanten berechnen"

Bei Wahl dieser Funktion werden die Konstanten des gewählten Zustandsreglers nach der Methode der Polvorgabe berechnet.

7.4.11 Funktion "Simulation durchführen"

Nach Wahl dieser Funktion erfolgt die Abfrage, ob die Simulationswerte gespeichert werden sollen. Wird die Abfrage positiv beantwortet, so sind der Dateiname und der Zeitabstand der zu

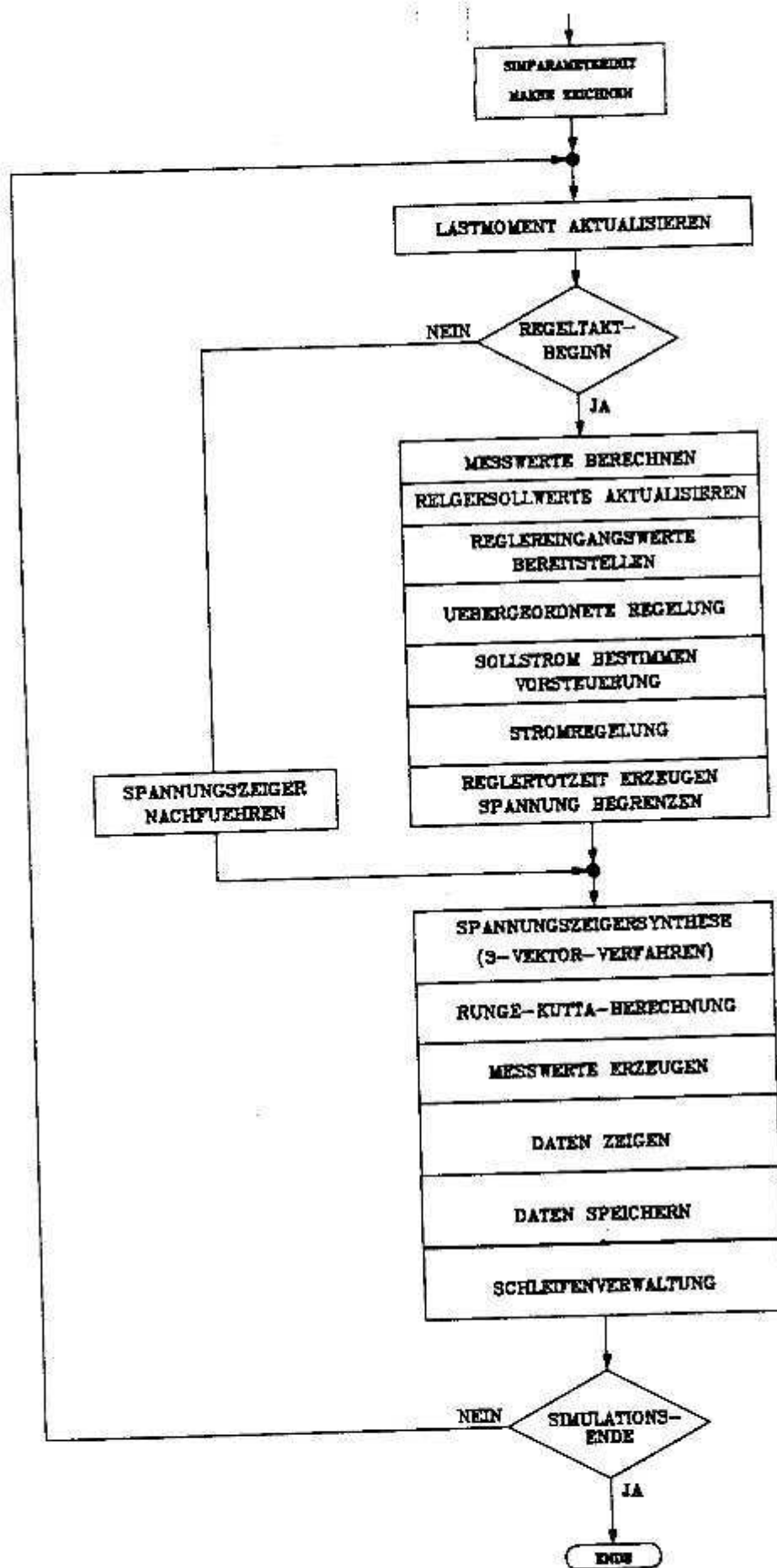


Bild 7.2 Vereinfachtes Flußdiagramm des Moduls Simulation

speichenden Werte einzugeben. Um das irrtümliche Überschreiben von existierenden Simulationsdateien zu verhindern, wird nach Eingabe des Dateinamens geprüft, ob die Datei bereits existiert. Ist dies der Fall, so erfolgt eine entsprechende Meldung und die Abfrage, ob die Datei überschrieben werden soll. Wird während der Durchführung der Simulation eine Taste gedrückt, so erfolgt die Abfrage, ob die Simulation beendet werden soll. Ist die vorgegebene Schrittzahl erreicht, so erfolgt die Abfrage, ob die Simulation fortgesetzt werden soll. Wird diese Abfrage positiv beantwortet, so ist vom Benutzer die Schrittzahl einzugeben, bis zu der die Simulation fortgesetzt werden soll.

7.4.12 Funktion "Programmende"

Bevor das Programm beendet wird, wird geprüft, ob eine Änderung der Simulationsparameter erfolgt ist. Ist dies der Fall, so erfolgt die Abfrage, ob die Simulationsparameter gespeichert werden sollen.

7.5 IMPLEMENTIERTE STROMREGLER

Bei allen Stromreglern ist der Spannungszeiger $\underline{u}^s$ im statorfesten System die Ausgangsgröße und $\underline{i}_{s011}^R$ eine der Eingangsgrößen. Bei den Stromreglern, deren Reglergesetz im rotorfesten System formuliert ist, ist aus $\underline{i}^s$ zunächst $\underline{i}^R$ zu berechnen und aus $\underline{u}^R$ dann $\underline{u}^s$ zu bestimmen. Diese Umrechnungen sind hier jeweils nicht angegeben.

In einigen Reglergesetzen tritt der Term $\underline{i}_k^R$ auf. Es kann hier gewählt werden, ob für diesen Term der Iststrom $\underline{i}^R$ oder der Sollstrom $\underline{i}_{s011}^R$ eingesetzt wird.

Im folgenden ist zu jedem Regler das verwendete Reglergesetz angegeben.

1. Stromsteuerung

$$\underline{u}^R = \underline{i}_{s011}^R + j\omega \cdot \underline{i}_k^R + j\omega$$

2. P-Stromregler mit Störgrößenkompensation

$$\underline{u}^R = k_1 * (\underline{i}_{s011}^R - \underline{i}^R) + j\omega * \underline{i}_k^R + j\omega$$

3. PI-Stromregler mit Störgrößenkompensation

$$\underline{I} := \underline{I} + \underline{i}_{s011}^R - \underline{i}^R$$

$$\underline{u}^R = k_1 * (\underline{i}_{s011}^R - \underline{i}^R) + k_2 * \underline{I} + j\omega * \underline{i}_k^R + j\omega$$

4. P-Stromzustandsregler mit Störgrößenkompensation

$$\underline{u}^R = k_1 * (\underline{i}_{s011}^R - \underline{i}^R) + (1+j\omega) * \underline{i}_k^R + j\omega$$

5. PI-Stromzustandsregler mit Störgrößenkompensation

$$\underline{I} := \underline{I} + \underline{i}_{s011}^R - \underline{i}^R$$

$$\underline{u}^R = k_1 * (\underline{i}_{s011}^R - \underline{i}^R) + k_2 * \underline{I} + (1+j\omega) * \underline{i}_k^R + j\omega$$

6. Stromzustandsregler mit Störgrößenkompensation und Totzeitberücksichtigung im rotorfesten System

$$\underline{u}^R := k_1 * \underline{i}_{s011}^R - k_2 * \underline{i}^R - k_3 * \underline{u}^R + j\omega * \underline{i}_k^R + j\omega$$

$$\text{mit } k_3 = e^{-T} , k_2 = k_3^2 / (1 - k_3) , k_1 = 1 / (1 - k_3)$$

7. Stromzustandsregler mit Störgrößenkompensation und Totzeitberücksichtigung im statorfesten System

Die Größen $\underline{i}_{s011}^R$ und $\underline{u}_p^R = j\omega$ sind zunächst ins statorfeste System umzurechnen. Im diesem System lautet dann das Reglergesetz:

$$\underline{u}^S := k_1 * \underline{i}_{s011}^S - k_2 * \underline{i}^S - k_3 * \underline{u}^S + \underline{u}_p^S$$

$$\text{mit } k_3 = e^{-T} , k_2 = k_3^2 / (1 - k_3) , k_1 = 1 / (1 - k_3)$$

Die nachfolgend beschriebenen Regler sind keine Stromregler, sondern Lagezustandsregler, die ohne unterlagerten Stromregler

arbeiten. Diese passen nicht in das Konzept des Programms, das stets einen Regler mit unterlagertem Stromregler voraussetzt. Deshalb wurden diese Lagezustandsregler den Stromreglern zugeordnet. Ist einer der beschriebenen Lagezustandsregler gewählt, so bleibt der übergeordnete Regler wirkungslos.

8. P-Lagezustandsregler mit Störgrößenkompensation

$$\underline{u}^R = j * (k_0 * (\varepsilon_{soll} - \varepsilon) + k_1 * (\varepsilon - \varepsilon_0) - k_2 * W - k_3 * i^R_q) + \\ + jw * \underline{i}^R + jw$$

9. PI-Lagezustandsregler mit Störgrößenkompensation

$$I := I + \varepsilon_{soll} - \varepsilon$$

$$\underline{u}^R = j * (k_0 * (\varepsilon_{soll} - \varepsilon) + k_1 * (\varepsilon - \varepsilon_0) - k_2 * W + k_3 * I - k_4 * i^R_q) + \\ + jw * \underline{i}^R + jw$$

7.6 IMPLEMENTIERTE ÜBERGEORDNETE REGLER

Im folgenden ist zu jedem wählbaren Regler das verwendete Reglergesetz angegeben.

1. Konstantes Moment

Dem Stromregler wird hier das Sollmoment m_{soll} vorgegeben.

Die Wahl dieses Reglers kann somit zum Testen des Verhaltens des gewählten Stromreglers dienen. Für das Sollmoment m_{soll} kann eine Zeitfunktion vorgegeben werden.

2. P-Drehzahl-Zustandsregler

$$m_{soll} = k_1 * w_{soll} - k_2 * W - k_3 * i_q$$

3. PI-Drehzahl-Zustandsregler

$$I := I + (w_{soll} - w)$$

$$m_{soll} = k_1 * w_{soll} - k_2 * w - k_3 * i_q + k_4 * I$$

4. P-Lage-Zustandsregler

$$m_{soll} = k_1 * (\varepsilon_{soll} - \varepsilon) + k_2 * (\varepsilon - \varepsilon_0) - k_3 * w - k_4 * i_q$$

5. PI-Lage-Zustandsregler

$$I := I + (\varepsilon_{soll} - \varepsilon)$$

$$m_{soll} = k_1 * (\varepsilon_{soll} - \varepsilon) + k_2 * (\varepsilon - \varepsilon_0) - k_3 * w - k_4 * i_q + k_5 * I$$

6. kaskadierter Lageregler mit Integralanteil

$$w_{soll} = k_1 * (\varepsilon_{soll} - \varepsilon) \quad I := I + (w_{soll} - w)$$

$$m_{soll} = k_2 * (w_{soll} - w) - k_4 * i_q + k_5 * I$$

7. zeitoptimaler Lageregler

Dieser Regler berechnet bei jedem Reglertakt den Anhalteweg. Unter Anhalteweg wird dabei der vom Rotor bis zum Stillstand bei ungünstigstem Lastmoment zurückgelegte Weg verstanden. Ist der Anhalteweg größer als die Differenz zum Lagesollwert, so wird das maximal erzeugbare Bremsmoment geschaltet. Ist der Anhalteweg kleiner als die Differenz zum Lagesollwert, so wird das maximal erzeugbare Beschleunigungsmoment geschaltet. Der Regler arbeitet somit als Zweipunktregler. Durch Einfügung einer neutralen Zone, in der das Moment 0 geschaltet wird, kann die Struktur des Reglers erweitert werden.

7.7 DATENSTRUKTUR DES PROGRAMMS

Das Programm verwaltet eine Vielzahl von Variablen.

Die Eingabe von Parametern erfolgt in Absolutwerten. Die Simulation selbst erfolgt in bezogenen Größen. Die Ausgabe der Simulationswerte auf dem Bildschirm bzw. in eine Datei erfolgt wahlweise in absoluten Größen oder in bezogenen Größen. Die Namen von bezogenen Variablen sind durch ein "b" als letzten Buchstaben des Namens gekennzeichnet.

Werden Simulationläufe gespeichert, so werden vom Programm zwei Dateien erstellt. Die Datendatei mit dem Suffix "DAT" enthält dabei nur die Daten der gespeicherten Größen im REAL-Format. Die Parameterdatei mit dem Suffix "PAR" enthält die Informationen, die zur Interpretation der Daten erforderlich sind. Dies sind z.B. die Anzahl der Daten eines Datensatzes, die Namen und Einheiten der Größen und der zeitliche Abstand der Werte. Darüber hinaus enthält sie auch die wesentlichen Simulationsparameter. Die Parameterdatei ist als "file of string[20]" organisiert. Die Auswertung dieser beiden Dateien erfolgt mit dem Grafikprogramm GREGA.

Im folgenden soll die Funktion einiger wichtiger Variablen erläutert werden:

- Schrittzahl: Anzahl der PWM-Schritte nach dem 3-Vektor-Verfahren
- Reglertakt: Länge eines Reglertakts in PWM-Schritten
- Zeigzaehler: Zähler für den Zeittakt bei der Bildschirmausgabe
- Speicherzaehler: Zähler für den Zeittakt bei der Dateiausgabe

7.8 DER MODUL SIMULATION

Bild 7.2 zeigt das vereinfachte Flußdiagramm des Moduls Simulation. Einige der gezeigten Blöcke können abhängig von den gewählten Parametern übersprungen werden.

Vor dem Eintritt in die Simulationsschleife wird die Initialisierung der Simulationsvariablen und der Aufbau der Bildschirmmaske durchgeführt.

Die Simulation erfolgt nach dem RUNGE-KUTTA-Algorithmus (RK-

Algorithmus) in bezogenen Größen im statorfesten System. Die Benutzung des statorfesten Systems ist deshalb sinnvoll, weil die verwendeten Spannungszeiger im statorfesten System konstant sind. Eingangs- und Ausgangsgrößen des RK-Algorithmus sind die Zustandsgrößen des Systems, die als Array organisiert sind. Die vorgegebene Schrittweite wird so an die Pulsmodulationstaktzeit (PMT) angepasst, daß die PMT ein ganzzahliges Vielfaches der Schrittweite ist. Bei der Realisierung des RK-Algorithmus ist zu gewährleisten, daß die Änderungszeitpunkte der Stellgröße mit Anfangszeitpunkten der Schritte zusammenfallen. Dies kann auf zwei Arten erfolgen:

- Durch Einfügung von Zwischenschritten bei Schritten, während denen eine Änderung der Stellgröße erfolgt
- Durch Schrittweitenanpassung

Die Einfügung eines Zwischenschritts kann zu einer sehr kleinen Schrittweite führen. Bei einer solchen sehr kleinen Schrittweite können - bedingt durch die begrenzte Genauigkeit der numerischen Rechnung - ungenaue Werte entstehen.

Da auch Daten, die während der Durchführung eines PWM-Taktes entstehen, auf dem Bildschirm dargestellt bzw. gespeichert werden sollen, führt die Schrittweitenanpassung zu unterschiedlichen Zeitabständen der Daten.

Aus diesen Gründen wurde ein kombiniertes Verfahren gewählt. Dieses fügt Zwischenschritte ein. Ist die dabei entstehende Schrittweite kleiner als der vorgegebene Minimalwert, so wird anstelle der Zwischenschritteinfügung eine Schrittweitenanpassung vorgenommen. Die Daten aus eingefügten Schritten werden bei der Weiterverarbeitung (Bildschirmausgabe, Speicherung) übergangen. Nach Abarbeitung des RUNGE-KUTTA-Algorithmus werden die Werte der Zustandsgrößen einzelnen Variablen zugewiesen. Die so gegebene Möglichkeit, den Variablen selbstdokumentierende Namen zu geben, trägt zur Übersichtlichkeit des Programms bei.

Aus den Größen im statorfesten System werden anschließend die Größen im rotorfesten System berechnet.

Die Aktualisierung des Lastmoments bei Vorgabe eines Lastmomentprofils erfolgt bei jedem PWM-Takt. Die Aktualisierung des Sollwerts bei Vorgabe eines Momenten-, Drehzahl- oder Lageprofils erfolgt bei jedem Reglertakt.

Die Reglertotzeit wird durch Verzögerung der Werte $u_{s11,d}$ und

us. 11, q um die Reglertaktzeit realisiert. Bei der Bestimmung der Steuerfaktoren nach dem 3-Vektor-Verfahren wird die einstellbare Stellgrößenauflösung berücksichtigt.

8. IMPLEMENTIERUNG DER REGELUNG

8.1 DER PROZESSOR 8096

Der Intel-Prozessor 8096 ist ein 16-Bit-Prozessor, der über Zusatzfunktionen verfügt, die ihn für Steuerungsaufgaben in Echtzeitanwendungen besonders geeignet machen.

Der Prozessor verfügt über 256 Register im Word-Format. Von diesen Registern besitzen die ersten 24 besondere Funktionen. Sie werden deshalb als "SPECIAL FUNCTION REGISTER" bezeichnet. Die übrigen Register sind frei benutzbar. Die große Zahl von gleichberechtigten Registern ist günstig hinsichtlich der Effektivität des Programmcodes.

Von den vielfältigen Funktionen des Prozessors sollen hier nur diejenigen erwähnt werden, die vom erstellten Programm benutzt werden.

Der serielle Port (Full Duplex) des Prozessors wird zur Kommunikation mit dem Terminal benutzt, von dem aus die Steuerung des Programms erfolgt. Der Prozessor verfügt über die beiden Timerbausteine Timer 1 und Timer 2. Die High-Speed-Outputs des Prozessors ermöglichen in Verbindung mit einem wählbaren Timerbaustein das Setzen und Zurücksetzen von Signalen zu vorgegebenen Zeitpunkten.

Der Prozessor wird mit einer Frequenz von 12,5 MHz getaktet. Aus dem Prozessortakt wird durch Frequenzteilung ein Interruptsignal mit Taktfrequenz 3,9 kHz erzeugt. Der Interruptabstand beträgt somit 256 μ s. Die Taktfrequenz der Stellgrößenausgabe beträgt 7,8 kHz. Der Prozessor stellt die üblichen 16-Bit-Operationen zur Verfügung einschließlich der Multiplikation und der Division. Gleitkommarechnungen werden nur durch den Prozessorbefehl NORML unterstützt. Die Berechnungen erfolgen deshalb fast durchgängig im Festkommaformat.

8.2 DAS ENTWICKLUNGSSYSTEM SBE96

Das Entwicklungssystem SBE96 besteht aus einem Hardwaremodul und einem Softwareprogramm. Die Platine des Anwenders wird hier in Anlehnung an die in der Beschreibung des Entwicklungssystems verwendete Terminologie als Userprototyp bezeichnet. Anstelle des Prozessors wird beim Userprototyp der Adapter des Hardwaremoduls eingesetzt. Das Softwareprogramm wird im Personal Computer (PC) geladen. Der Hardwaremodul kommuniziert über eine serielle Schnittstelle mit dem PC. Der Hardwaremodul arbeitet mit dem Prozessor 8096 und führt somit eine Echtzeitemulation durch. Das Entwicklungssystem bietet vielfältige Features. Zu erwähnen sind hier die Möglichkeit zum Laden und Starten eines Programms im INTEL-OBJ-Format und die Möglichkeit zur Übertragung des Inhalts eines Adressbereichs zum PC und zur Abspeicherung der Daten in einer Datei.

Wählbar ist beim System der Operationsmode. Dieser bestimmt wie der Adressbereich des Prozessor 8096 von 8 kByte zwischen dem Hardwaremodul und dem Userprototyp aufgeteilt wird. Wichtig ist in diesem Zusammenhang, daß unabhängig vom gewählten Operationsmode bei einem Schreib- oder Lesevorgang die Adresse mit der zugehörigen Steuerinformation stets auch auf den Bus des Userprototyps geleitet wird. Der dem Hardwaremodul zugeordnete Speicherbereich darf also beim Userprototyp nicht bestückt sein.

Hier wird mit dem Operationsmode 5 gearbeitet. Dieser ordnet die Speicherbereiche von 100H bis 7FFH und 2000H bis 5FFFFH dem Hardwaremodul zu.

Der Speicherbereich von 6000H bis 7FFFFH muß dem verwendeten Userprototyp zugeordnet sein, da sich in diesem Adressbereich die zur Lage- und Strommessung verwendeten Bausteine befinden.

8.3 GRUNDSTRUKTUR DES ASSEMBLERPROGRAMMS

8.3.1 BEZEICHNUNGEN

Symbolische Namen von Speicherplätzen, Sprungadressen (Label) und Befehle des Assemblerprogramms werden in Großbuchstaben geschrieben. Unterschieden wird zwischen den Datentypen BYTE (8 Bit), WORD (16 Bit) und LONGWORD (32 Bit). Wird der in einem dieser Datentypen gespeicherte Wert als vorzeichenbehaftet angesehen, so werden die Bezeichnungen SHORT INTEGER, INTEGER und LONG INTEGER verwendet. Bei vorzeichenlosen Werten wird diese Tatsache dadurch gekennzeichnet, daß dem Datentyp das Adjektiv UNSIGNED vorangestellt wird.

Die Darstellung von Programmsequenzen erfolgt direkt in der Assemblersprache oder unter Verwendung von Bezeichnungen der Programmiersprache Pascal.

Bei der Beschreibung von Rechnungen im Festkommaformat werden die Begriffe trunc, div und mod verwendet.

8.3.2 PROGRAMMEBENEN

Programmebenen sind dadurch gekennzeichnet, daß der Wechsel zwischen den Ebenen beim Programmablauf zu Zeitpunkten erfolgt, die bei der Erstellung des Programms nicht bestimmbar sind.

Das zu beschreibende Programm besitzt drei Ebenen.

Die Ebene 1 besorgt dabei die Kommunikation mit dem als Steuerkonsole verwendeten Terminal und die Steuerung und Überwachung des Programmablaufs.

In der Ebene 2 (REGELUNG) werden die umfangreichen Berechnungen zur Regelung ausgeführt.

In der Ebene 3 (MOT_INT) erfolgt die Spannungszeigersynthese und die Stellgrößenausgabe.

Das Programm kommt dabei mit einer Interruptquelle aus.

Die hardwaremäßig realisierte Interruptquelle erzeugt Interruptsignale im Abstand von 256 µs. Die Realisierung von zwei Programmebenen mit einer Interruptquelle wird dabei im folgenden beschrieben.

Der Interrupt bewirkt den Aufruf der Routine MOT_INT. Bei jedem Interrupt wird der Zähler INT_COUNT decrementiert. Hat INT_COUNT den Wert 1 erreicht, so wird INT_COUNT mit dem Wert INT_ZAHL geladen und es erfolgt nicht der Aussprung aus der Interrupt-routine, sondern es wird die Routine REGELUNG aufgerufen.

Erst nach Abarbeitung der Routine REGELUNG erfolgt dann der Aussprung aus der Interruptroutine. Der Einsprung in die Routine REGELUNG darf aber nur dann erfolgen, wenn diese Routine bereits abgearbeitet und der Aussprung aus der Interruptroutine erfolgt ist. Die Prüfung erfolgt anhand des Bits 0 der Variablen STATUSFLAG. Das Bit 0 wird dabei beim Einsprung in die Routine REGELUNG gesetzt und vor dem Aussprung aus der Interruptroutine rückgesetzt.

Die zeitliche Aufeinanderfolge der Programmebenen ist in Bild 8.1 dargestellt.

Unterschieden werden kann zwischen zeiteffizienter Programmierung und speichereffizienter Programmierung.

Speichereffiziente Programmierung ist gekennzeichnet dadurch, daß viele Abschnitte des Programmcodes mehrfach genutzt werden. Dies führt im Programmcodes zum vermehrten Auftreten von Unterprogramm-aufrufen und zum vermehrten Auftreten von bedingten und unbedingten Sprüngen. Die Abarbeitung der zusätzlichen Befehle geht auf Kosten der Zeiteffizienz des Programms.

Zeiteffiziente Programmierung verzichtet dagegen auf die Mehrfachnutzung von Programmcodes. Dies führt zu einem erhöhten Speicherbedarf für den Programmcodes. Bei zeiteffizienter Programmierung kann der Programmieraufwand und die Länge des Quellcodes durch Verwendung von MACROS verringert werden.

Bei dem vorhandenen System ist sowohl auf Zeiteffizienz als auch auf Speichereffizienz zu achten. Es wurden deshalb in den verschiedenen Programmebenen unterschiedliche Programmierstile verwendet:

Die Ebene 1 (Kommunikation) ist vollständig speichereffizient programmiert, da die Rechenzeit hier unkritisch ist.

Die Ebene 3 (Stellgrößenausgabe) ist vollständig zeiteffizient

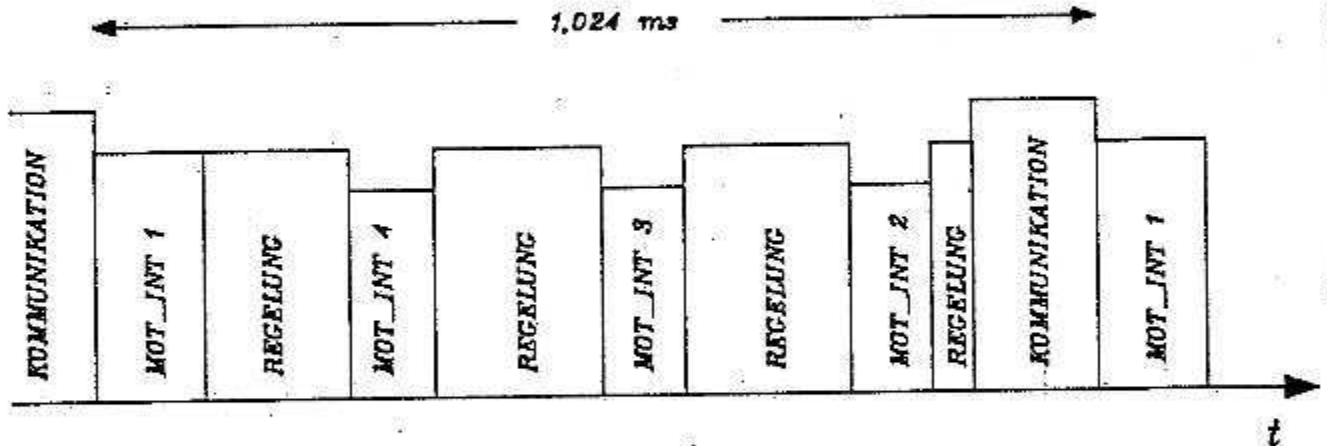


Bild 8.1 Zeitliche Aufeinanderfolge der Programmebenen

programmiert.

Bei der Ebene 2 (REGELUNG) wurde ein Kompromiß zwischen Zeiteffizienz und Speichereffizienz geschlossen.

Variabilität wurde hier erzielt, indem unter Benutzung der Möglichkeit der bedingten Assemblierung gesteuert vom Wert der Assemblervariablen MAKRO bestimmte Routinen sowohl als Unterprogramme als auch als MACROS verwirklicht werden können.

8.3.3 DIE DATENSTRUKTUR

Der Stack wird außerhalb des Registerfiles im Bereich von 100H bis 200H angesiedelt.

Dies verlängert die Ausführungszeit derjenigen Befehle, die den Stack benutzen. Jedoch erwies sich diese Maßnahme als unumgänglich, da die Speicherplätze des Registerfiles vollständig benötigt werden.

Die Register des Prozessors werden hier entsprechend ihrer Benutzung im Programm unterteilt in Register und Variable.

Die Register dienen dabei zur Aufnahme von Zwischenergebnissen. Die Wertübergabe zwischen einzelnen Routinen erfolgt nur in Ausnahmefällen in Registern.

Jeder Programmebene ist ein eigener Registersatz zugeordnet. Der Ebene 3 (MOT_INT) sind 4 Word-Register zugeordnet, die mit AX bis DX bezeichnet sind.

Die Ebene 2 (REGELUNG) verfügt über 8 Word-Register, die mit A1X bis H1X bezeichnet sind.

Fünf Word-Register, die mit A2X bis E2X bezeichnet sind, stehen der Kommunikationsebene (Ebene 1) zur Verfügung.

Die Word-Register sind unter Verwendung der EQU Instruktion jeweils in 2 Byte-Register unterteilt, die anstelle des X jeweils L bzw. H als letzten Buchstaben der Bezeichnung besitzen.

Variable sind im Gegensatz zu den Registern mit Namen bezeichnet, die auf ihre Verwendung hinweisen.

Besondere Variable sind Transformationsvariable. Sie dienen zur Wertübergabe an und Wertübernahme von Transformationsroutinen.

Zu nennen sind hier die Variablen YY,YY,DD,QQ,BET und PHA.

Die Verwendung der Transformationsvariablen ermöglicht es, daß die Transformationsroutinen sowohl zur Umrechnung von Strom- als auch von Spannungszeigern benutzt werden können.

Besondere Bedeutung kommt auch den Übergabevariablen UBET_REG und UPHAX_REG zu. Sie dienen zur Übergabe der Reglerausgangsgrößen an die Routine MOT_INT. Nachdem die Reglerausgangsgrößen berechnet worden sind, werden sie den Übergabevariablen UBET_REG und UPHAX_REG zugewiesen. Anschließend wird Bit 1 der Variablen STATUSFLAG gesetzt. Die Übernahme der Werte in die entsprechenden Variablen UBET und UPHAX der Routine MOT_INT erfolgt nur bei gesetztem Bit 1 der Variablen STATUSFLAG. Nach der Übernahme wird Bit 1 der Variablen STATUSFLAG zurückgesetzt.

8.3.4 DATENFORMAT DER VARIABLEN

Die Rechnungen erfolgen im Festkommaformat, da der Prozessor über keine Gleitkommabefehle verfügt. Der Aufbau von Gleitkommaroutinen aus den verfügbaren Prozessorbefehlen würde zu einer Rechenzeit führen, die in der hier beschriebenen Anwendung nicht akzeptabel wäre.

Bei allen Überlegungen zum Datenformat und zur Genauigkeit muß vom Wertebereich der digitalisierten Meßwerte und vom Wertebereich der digitalen Stellgrößen ausgegangen werden. Das Datenformat muß so gewählt werden, daß beide Wertebereiche in ihm darstellbar sind. Der Wertebereich der digitalen Stellgrößen sollte dabei nur einen kleinen Teil des durch das Datenformat gegebenen Wertebereichs ausschöpfen, damit die Werte nicht durch Überlauf verfälscht werden.

Beim hier betrachteten System werden die Rotorlage und die drei Strangströme gemessen.

Die Rotorlage wird durch einen Incrementalgeber erfasst. Dieser liefert 3600 Incremente pro elektrische Umdrehung. Der dem Incrementalgeber nachgeschaltete Lagezähler stellt die Lage als LONGWORD dar. Es kann somit ein Bereich von $1,19 \cdot 10^6$ Umdrehungen dargestellt werden.

Die Strangströme werden mittels Hallsonden gemessen. Die Ausgangsspannungen der Hallsonden steuern VCO's, deren Oszillatorfrequenzen über nachgeschaltete Zähler auf Zählerstände abgebildet werden. Es ergibt sich hier eine Empfindlichkeit von ca. 30 Incrementen pro Ampere. Bei Zugrundelegung eines Maximalstroms von 10 A ergibt sich für die digitalisierten Strommesswerte ein Wertebereich von - 300 bis +300 Incrementen.

Die Stellgröße u ist hier ein komplexer Spannungszeiger. Dieser wird im System durch zwei in digitaler Form vorliegende Steuerzeiten t_1 und t_2 repräsentiert. Dabei gilt - vorgegeben durch die Hardware - für den Wertebereich der Steuerzeiten:

$$0 \leq t_1 \leq t_2 \leq 128$$

Der Wertebereich der Strommesswerte erfordert das INTEGER-Datenformat für die Festkommarechnungen. Bei Verwendung des INTEGER-Formats kann mit den Werten t_1 und t_2 noch das 256-fache der maximal möglichen Stellgröße korrekt dargestellt werden. Ausgehend hiervon wurde das INTEGER-Format für die Festkommarechnung zugrundegelegt.

8.3.5 RECHNUNGEN IM FESTKOMMAFORMAT

Das Rechnen bei Verwendung des Festkommaformats erfordert einige grundsätzliche Überlegungen, die hier kurz dargestellt werden sollen.

Bei Addition und Subtraktion muß verhindert werden, daß das Ergebnis durch Überlauf falsch wird. Diese Prüfung auf Überlauf kann nach der Operation durch Abfragen des OVERFLOW-Flags erfolgen. Günstiger ist es jedoch, die Operanden vor der Rechenoperation so zu begrenzen, daß kein Überlauf auftreten kann.

Bei Multiplikation von zwei INTEGER-Werten besitzt das Produkt LONGINTEGER-Format. Anschließend an die Multiplikation ist zu prüfen, ob das Produkt ins INTEGER-Format paßt. Ist dies nicht der Fall, so ist eine Begrenzung des Produkts auf den entsprechenden Maximalwert erforderlich.

Bei der Division besitzen der Dividend A LONGINTEGER-Format und der Divisor B und der Quotient C INTEGER-Format. Hieraus folgt, daß bei einer vorzeichenlosen Division (Befehl DIVU) zur Vermeidung eines Überlaufs die Bedingung:

$$A < B * 2^{16} \quad (8.3.4.1)$$

erfüllt sein muß.

Bei einer vorzeichenbehafteten Division (Befehl DIV) muß gelten:

$$|A| < |B| * 2^{15} \quad (8.3.4.2)$$

Bei Regelalgorithmen tritt häufig die Addition von Produkten in der Form $A*B + C*D$ auf. Wurde mindestens ein Produkt begrenzt, so tritt insbesondere bei ungleichem Vorzeichen der Produkte eine kritische Verfälschung des Resultats auf.

Durch eine Umformung des Ausdrucks kann in bestimmten Fällen das Auftreten eines Überlaufs verhindert werden, wie im folgenden gezeigt wird.

Ist der Ausdruck $A*B - C*D$ mit $A, C > 0$ und kleiner Differenz $A-C$ zu berechnen, so kann durch Benutzung des umgeformten Ausdrucks:

$$A*(B-D) + (A-C)*D$$

eine Verbesserung des Überlaufverhaltens erreicht werden, da jetzt nur beim ersten Summand eine Begrenzung erforderlich sein kann.

Besondere Aufmerksamkeit ist den zur Berechnung von nichtlinearen Funktionen verwendeten Tabellen zu schenken. Sei $y = f(x)$ eine Funktion, deren Argumente x und Werte y in einem der betrachteten Datenformate darstellbar sind. Bei der Tabellierung wird als Eingangswert (Adressoffset) der Wert $x \cdot 2^{-n}$ und als Tabellenwert (Adressinhalt) der Wert $y \cdot 2^{-m}$ benutzt. Durch den Exponenten n wird die Anzahl der Tabelleneinträge bestimmt, während durch den Exponenten m das Format der Tabelleneinträge festgelegt wird. Die durch die Tabelle realisierte Funktion sei f_r bezeichnet. Der Fehler:

$$\Delta y = f_r(x) - f(x)$$

setzt sich dann zusammen aus einem Eingangsfehler, der durch die Anzahl der Tabelleneinträge bestimmt wird, und einem Ausgangsfehler, der durch das Format der Tabelleneinträge bestimmt wird:

$$\Delta y = \Delta_e y + \Delta_a y$$

Der maximale Eingangsfehler in Abhängigkeit von $x = \text{int}(x/2^n) \cdot 2^n$ ist gegeben durch:

$$\Delta_e y = f(x + 2^n) - f(x)$$

Der angegebene Eingangsfehler kann durch entsprechende Rundung des Arguments halbiert werden.

Der maximale Ausgangsfehler in Abhängigkeit von $y = \text{int}(y/2^m) \cdot 2^m$ ist gegeben durch

$$\Delta_a y = y + 2^m - y = 2^m$$

Der angegebene Ausgangsfehler halbiert sich, wenn die tabellierten Werte gerundet sind.

Ein günstiges Verhältnis zwischen der erzielten Genauigkeit und der Tabellenlänge wird erzielt, wenn $\Delta_e y \approx \Delta_a y$ ist.

8.3.6 DATENFORMAT DER REGLERKONSTANTEN

Bei den verwendeten multiplikativen Reglerkonstanten ist die Darstellung in einem Gleitkommaformat erforderlich.

Dieses besteht aus dem WORD KONST und dem vorzeichenlosen BYTE NEX. Der Konstante wird der Wert $KONST \cdot 2^{-NEX}$ zugeordnet. Ist k die darzustellende Konstante, so wird die Darstellung der Konstante am genauesten, wenn gewählt wird:

$$NEX = \text{int}(\log_2 ((2^{15}-1)/k))$$

Die Programmsequenz zur Multiplikation der Zahl B1X mit der Konstanten lautet:

```
MUL      B1X, KONST
LD        A1L, NEX
SHRAL     B1X, A1L
```

Das Laden des Registers A1L mit dem Wert NEX ist hier erforderlich, da die Konstanten außerhalb des Registerfiles angeordnet sind. Darstellbar sind also Konstanten, deren Betrag im Bereich 2^{-15} bis $2^{15}-1$ liegt.

Ist damit zu rechnen, daß das Resultat der Multiplikation die zulässige Dualstellenzahl überschreitet, so muß anschließend eine diesbezügliche Prüfung und gegebenenfalls Begrenzung des Resultats erfolgen.

Bei der Berechnung der Integralanteile in Reglergesetzen ist die Multiplikation eines LONGINTEGER-Werts mit einer Reglerkonstanten k erforderlich. Eine Möglichkeit des Vorgehens besteht hier darin, den LONGINTEGER-Wert vor der Multiplikation auf das INTEGER-Datenformat zu begrenzen. Der Betrag des Produkts ist dann stets $\leq 2^{15} \cdot |k|$. Ist nun $|k| \ll 1$, so kann eine starke Verfälschung des Resultats auftreten.

Eine Verbesserung kann hier dadurch erzielt werden, daß der Exponent NEX der Konstante k in zwei Anteile zerlegt wird:

$$NEX = NEX1 + NEX2$$

Dabei wird NEX2 so gewählt, daß $KONST \cdot 2^{-NEX2} \approx 1$ ist.

Die Programmsequenz zur Multiplikation des in den Register B1X und C1X stehenden LONGINTEGER-Werts mit der Konstanten k lautet dann:

```
LD      A1L,NEX1
SHRAL   B1X,A1L
LONGINTEGER auf INTEGER begrenzen
MUL     B1X,KONST
LD      A1L,NEX2
SHRAL   B1X,A1L
LONGINTEGER auf INTEGER begrenzen
```

Da das Resultat nun den durch das INTEGER-Format gegebenen Wertebereich ausschöpft, wird die Verfälschung des Resultats vermieden.

8.4 DIE INTERRUPTROUTINE MOT_INT

8.4.1 DIE STRUKTUR DER INTERRUPTROUTINE

Nach dem Einsprung in die Interruptroutine wird der Zählerstand des Timers 1 in die Variable ZEIT übernommen. Der Wert ZEIT dient als Referenzwert für die Berechnung der Steuerzeitpunkte.

Die weiteren Aktionen erfolgen abhängig vom Byte-Zähler INT_COUNT. Dieser Zähler ist 1 aus 8 kodiert. Dieser Kodierung ermöglicht zählerstandabhängige Sprünge durch den Befehl

```
JBS     INT_COUNT,x,LABEL
```

Bei einem normal codierten Zähler sind hierfür zwei Befehle notwendig:

```
CMPB    INT_COUNT,#x
JE       LABEL
```

Bei der Beschreibung des Programms wird der Zähler INT_COUNT zur Vereinfachung der Darstellung jedoch im folgenden als normal kodierter Zähler betrachtet.

Beim Zählerstand 2 werden die Stromzähler mit ihrem Startwert geladen. Beim Zählerstand 1 erfolgt das Latchen des Lagezählers und der Stromzähler. Anschließend wird die Stellgrößenausgabe mit der Routine CAMFILL durchgeführt. Der weitere Ablauf ist wieder zählerstandsabhängig. Ist INT_COUNT > 1, so erfolgt das Nachführen des Spannungszeigers und die Durchführung der Spannungs-

zeigersynthese (SPSYNTH), wenn die Option "Nachführen" gewählt ist. Anschließend erfolgt der Aussprung aus der Interruptroutine. Ist INT_COUNT = 1, so werden die Reglerausgangswerte übernommen. Stehen keine neuen Reglerausgangswerte zur Verfügung, so wird der Spannungszeigerwinkel UPHAX um den Wert USPEED erhöht. Dies ermöglicht die Realisierung von Drehfeldern bei ausgeschalteter Regelung.

Anschließend erfolgt dann die Spannungszeigersynthese (SPSYNTH), das Laden des Zählers INT_COUNT mit dem Inhalt der Speicherstelle INT_ZAHL und der Einsprung in die Routine REGELUNG, wenn diese bereits abgearbeitet ist.

Bild 8.2 zeigt das vereinfachte Flußdiagramm der Interruptroutine MOT_INT.

8.4.2 DIE ROUTINE SPSYNTH

Diese Routine führt die Spannungszeigersynthese durch. Eingangsvariable sind: UBET, UPHAX

Ausgangsvariable sind: ZEIT1, ZEIT2, ZEIT3, ZEIT4

Sind ZEIT1 und ZEIT2 nach dem 3-Vektor-Verfahren berechnet worden, so erfolgt die Anpassung an Taktzeit 128 μ s der Stellgrößenausgabe durch die Programmsequenz:

```
AX      := ZEIT1 div 2;  
ZEIT3   := ZEIT1 - AX;  
ZEIT1   := AX;  
AX      := ZEIT2 div 2;  
ZEIT4   := ZEIT2 - AX;  
ZEIT2   := AX;
```

Während eines Interrupts erfolgt eine zweimalige Stellgrößenausgabe. Bei der ersten Ausgabe werden die Werte ZEIT1 und ZEIT2 verwendet, während bei der zweiten Ausgabe die Werte ZEIT3 und ZEIT4 benutzt werden. Diese Implementierung gewährleistet, daß die Auflösung der Stellgröße durch die zweimalige Stellgrößenausgabe nicht reduziert wird.

8.4.3 DIE ROUTINE CAMFILL

Die diese Routine bewirkt die Stellgrößenausgabe an den Motor.

Eingangsvariable sind: ZEIT, ZEIT1, ZEIT2, ZEIT3, ZEIT4

Die Vorgehensweise ist dabei so, daß die Steuerinformation zusammen mit dem Timerstand, bei dem die Ausgabe der Steuerinformation erfolgen soll, in das 8 Speicherplätze umfassende CAM des Prozessors geschrieben werden. Ist der angegebene Timerstand erreicht, so erfolgt die Ausgabe der Steuerinformation an die High-Speed-Outputs und der betreffende CAM-Platz steht zur Aufnahme von Information erneut zur Verfügung.

Hier werden zwei der High-Speed-Output benutzt. Diese bewirken das Enable bzw. Umschalten eines Multiplexers. Ist der Multiplexer disabled, so sind seine Ausgänge alle im Low-Zustand. Die Ausgangssignale des Multiplexers werden zur Steuerung der drei Halbbrücken des Leistungsteils benutzt. Das Eingangssignal des Multiplexers wird dabei vom I/O-Port 1 des Prozessors geliefert. Das so realisierte 3-Vektor-Verfahren erfordert das Einschreiben von 3 Steuerinformationen in das CAM pro Taktperiode.

Bei der Bestimmung der Zeitpunkte, zu denen eine Steuerinformation ausgegeben werden soll, ist die Zeitdauer zu berücksichtigen, die zur Abarbeitung der Laderoutine erforderlich ist. Erfolgt dies nicht, so ist es möglich, daß eine Steuerinformation erst nach dem Abarbeitungszeitpunkt in das CAM aufgenommen wird. In diesem Fall verbleibt die Steuerinformation im CAM bis der Timer wieder den vorgesehenen Ausgabezeitpunkt erreicht. Dies führt zu einer Verstopfung des CAM mit katastrophalen Folgen für den Programmablauf.

8.5 DIE ROUTINE REGELUNG

8.5.1 DIE STRUKTUR DER ROUTINE

Diese Routine führt in der Reihenfolge der Aufzählung folgende Funktionen durch:

1. Auslesen der Stromzählerstände in die Variablen STROM1, STROM2 und STROM3
2. Auslesen des Lagezählerstands in die LONGWORD-Variable NEULAGE
3. Neuberechnung der Rotordrehzahl SPEED, wenn bestimmte Bedingungen erfüllt sind.
4. Berechnung der Werte EL_UMDR und EL_WINK_INT aus NEULAGE
5. Berechnung der Komponenten IX und IY des Stromzeigers im statorfesten System.
6. Berechnung der Komponenten ID und IQ des Stromzeigers im rotorfesten DQ-System mit der Routine XYDQ.
Berechnung des Winkels UPHAD_MESS aus dem Winkel UPHAX_MESS
7. Übergeordnete Regelung
Der Aufruf des Reglers erfolgt durch Verzweigung zu derjenigen Regleroutine, deren Adresse in der Variablen REG_ADR steht.
Reine Testroutinen verzweigen nach Abarbeitung zur Adresse REGEND.
Regleroutinen berechnen den Wert IQSOLL und übergeben diesen der Stromregleroutine. Hierzu wird ein Sprung zur Adresse STRREG durchgeführt.
8. Stromregelung (Adresse STRREG)
Nach der Begrenzung des Sollwerts IQSOLL erfolgt die Verzweigung zu derjenigen Stromregleroutine, deren Adresse in der Variablen SREG_ADR steht.
Die Stromregleroutinen berechnen die Werte UBET_REG und UPHAX_REG springen zur Adresse REGEND

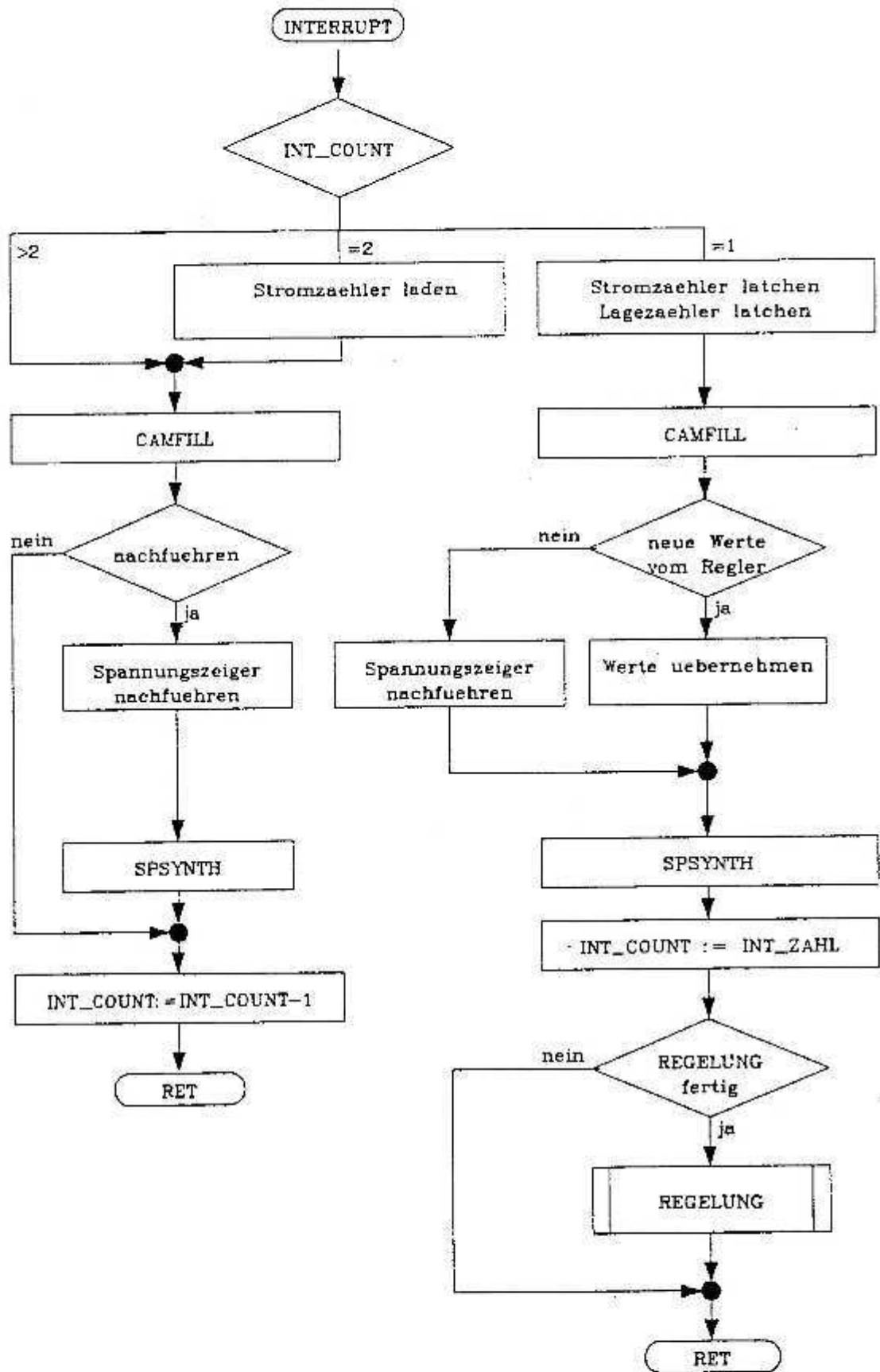


Bild 8.2 Vereinfachtes Flußdiagramm der Routine MOT_INT

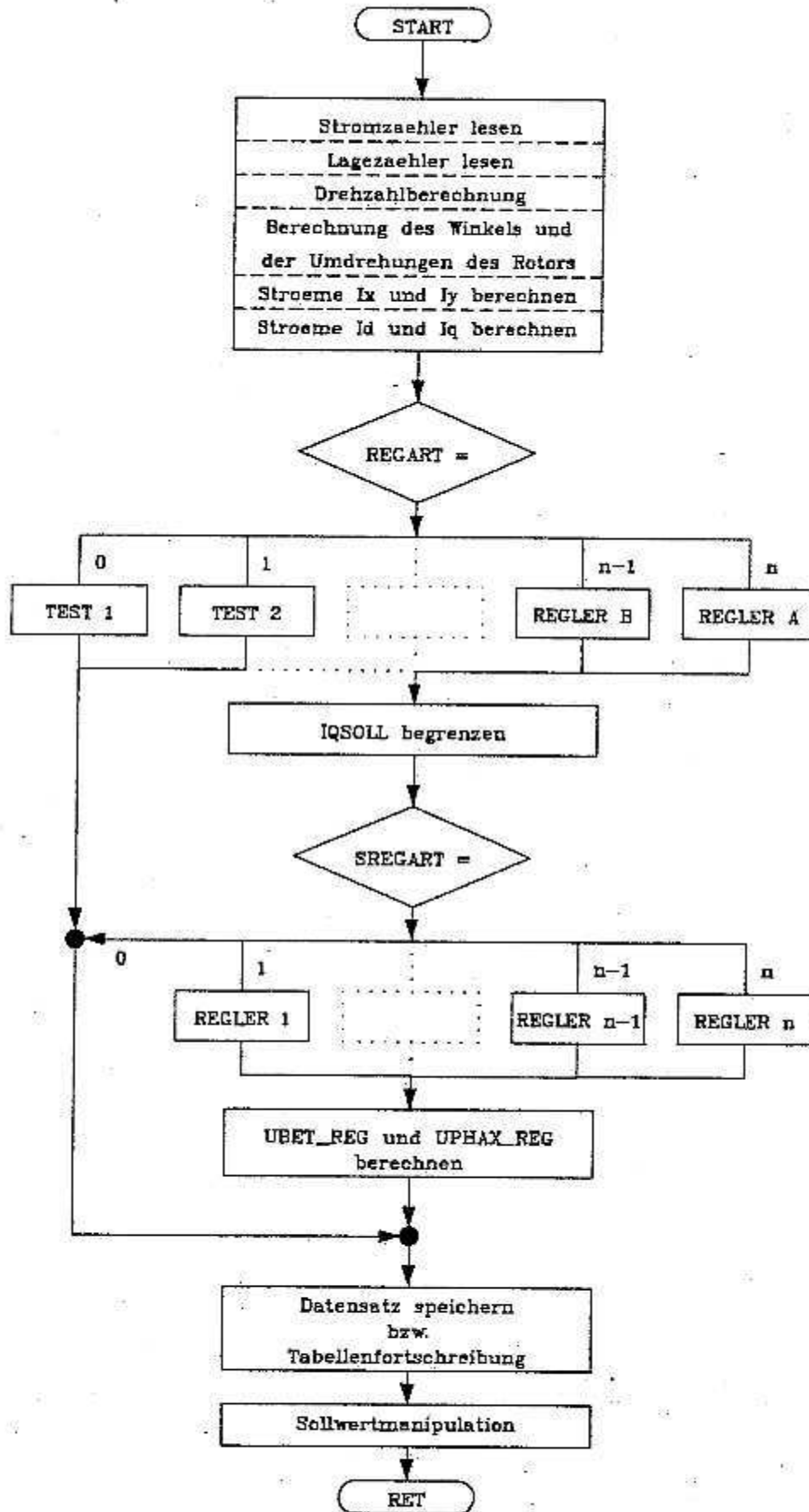


Bild 8.3 Vereinfachtes Flußdiagramm der Routine REGELUNG

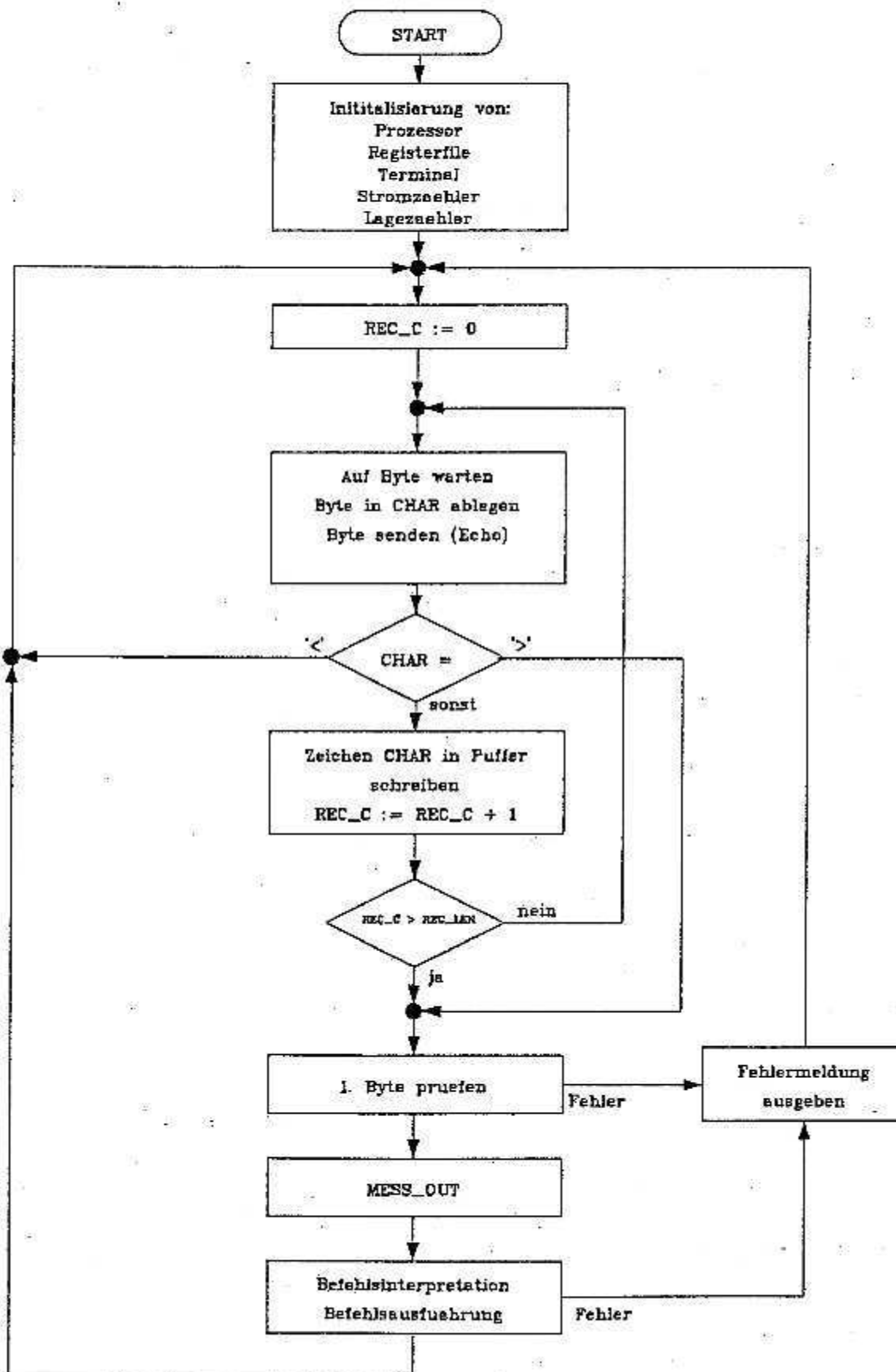


Bild 8.4 Vereinfachtes Flußdiagramm des Hauptprogramms

9. Speichern von Datensätzen (Adresse REGEND)

Wenn Bit 0 der Variablen AUFTRAG_FLAG gesetzt ist, so wird der gewählte Datensatz in den Speicherbereich des Systems übertragen. Ist Bit 1 der Variablen AUFTRAG_FLAG gesetzt, so erfolgt die Durchführung der Tabellenfortschreibung. Ein Datensatz besteht aus jeweils 22 Byte. Es kann zwischen 5 Datensatzarten gewählt werden. Die Datensatzart wird in der Variablen TAB_ART gespeichert.

10. Manipulation von Sollwerten

Es erfolgt die Verzweigung zur Routine, deren Adresse in der Variablen CNG_ADR steht. Die Variable CNG_ART enthält dabei die Nummer der gewählten Routine. Bei CNG_ART = 0 erfolgt direkt der Aussprung aus der Routine REGELUNG.

Die wählbaren Routinen führen Änderungen der Sollwerte durch. Implementiert sind hier Rechteckfunktionen mit wählbarer Periodendauer für Strom-, Drehzahl- und Lagesollwert. Ferner ist eine Dreieckfunktion mit wählbarer Periodendauer für den Lagesollwert vorhanden.

Weiterhin sind Routinen verfügbar, die den Drehzahl- bzw Lagesollwert ändern, sobald der Sollwert erreicht ist. Das Kriterium hierfür ist, daß die Differenz zwischen Sollwert und Istwert während einer vorgebbaren Zeitdauer kleiner als ein vorgebbarer Wert ist. Die implementierten Routinen enden sämtlich mit einem RET-Befehl und führen somit den Aussprung aus der Routine REGELUNG durch.

Ein vereinfachtes Flußdiagramm der Routine REGELUNG ist in Bild 8.3 dargestellt.

Die Routinen der übergeordneten Regelung bestehen aus Testroutinen und echten Regelroutinen.

8.5.2 Testroutinen

Die Testroutinen erlauben in Zusammenhang mit der durch den Kommandointerpreter gegebenen Möglichkeit zum Vorgeben eines Spannungszeigers und zur Definition eines Drehfelds die Prüfung der Systemfunktionen und die Einstellung der Systemkonstanten. Folgende Testroutinen sind implementiert:

1. AUSWERTUNG:

Diese Routine berechnet aus den Messwerten zusätzliche Werte.

2. MODELLO_TEST:

Diese Routine berechnet die Werte UX_KOMP und UY_KOMP nach den Formeln:

$$UX_KOMP = UZKFAKT * (IX - SPEED_MTEST * LKONST * IY)$$

$$UY_KOMP = UZKFAKT * (IY + SPEED_MTEST * LKONST * IX)$$

Der Wert SPEED_MTEST wird dabei bei der Definition des Drehfelds entsprechend der Frequenz des Drehfelds vorgegeben. Die Berechnung erfolgt im statorfesten System. Die Routine gestattet durch Vergleich der Werte UBET_KOMP, UBET und UPHAX_KOMP, UPHAX die Bestimmung der Konstanten UZKFAKT und LKONST. Zunächst wird dabei UZKFAKT bei SPEED_MTEST = 0 bestimmt. Anschließend erfolgt die Bestimmung von LKONST. Bei der Bestimmung der Konstanten muß der Rotor durch die elektromechanische Bremse fixiert sein.

3. MODELL1_TEST:

Die Werte UD_KOMP und UQ_KOMP werden hier nach den Formeln:

$$UD_KOMP = UZKFAKT * (ID - SPEED_MTEST * LKONST * IQ)$$

$$UQ_KOMP = UZKFAKT * (IQ + SPEED_MTEST * (LKONST * ID + EKONST))$$

berechnet.

Die Berechnung erfolgt hier im rotorfesten System. Neben der beschriebenen Bestimmung der Konstanten UZKFAKT und LKONST gestattet diese Routine die Bestimmung der Konstante EKONST, wenn die Konstanten UZKFAKT und LKONST bekannt sind. Zur Bestimmung der Konstanten EKONST muß die Bremse gelöst sein.

4. MODELL2_TEST:

Diese Routine berechnet die Werte UD_KOMP und UQ_KOMP nach den Formeln:

$$UD_KOMP = UZKFAKT * (ID - SPEED * LKONST * IQ)$$

$UQ_KOMP = UZKFART * (IQ + SPEED * (LKONST * ID + EKONST))$

Der Unterschied zur Routine MODELL1_TEST besteht darin, daß hier anstelle des konstanten Werts SPEED_MTEST der Wert SPEED benutzt wird. Abweichungen vom stationären Zustand infolge von rotorwinkelabhängigen Reibmomenten und Polradpendelungen werden also bei Verwendung dieser Routine berücksichtigt.

8.5.3 LAGE- UND DREHZAHLEGLER

Alle übergeordneten Regler arbeiten mit dem gleichen Gleitkomma-konstantensatz. Dieser aus 4 LONGWORD-Werten bestehende Konstantensatz bildet die erste Zeile des Konstantenarrays. Mit einem Befehl "G" des Kommandointerpreters läßt sich eine wählbare Zeile des Konstantenarrays in die erste Zeile laden oder die erste Zeile in eine wählbare Zeile laden. Diese Implementierung ermöglicht es, die zu verschiedene Reglereinstellungen und verschiedenen Reglern gehörenden Konstanten verfügbar zu haben. Die Reglereinstellung kann dann einfach durch Laden des Konstantensatzes erfolgen. Im folgenden wird für die implementierten Algorithmen jeweils das Reglergesetz angegeben. Die Konstanten sind dabei mit K0 bis K3 bezeichnet. Die Verwendung des Zuweisungszeichens "==" anstelle des Gleichheitszeichens "=" weist auf den rekursiven Charakter eines Gesetzes hin.

1. P-Drehzahlzustandsregler

$IQSOLL = K0 * (SOLLSPEED - SPEED) + K2 * SOLLSPEED$

Der Term $K2 * SOLLSPEED$ stellt dabei regelungstechnisch eine Vorsteuerung zum Ausgleich des Reibmoments dar.

2. PI-Drehzahlzustandsregler

Hier tritt im Reglergesetz noch ein Integralterm hinzu:

$INTEGRAL := INTEGRAL + SOLLSPEED - SPEED$

$$IQSOLL = K0*(SOLLSPEED - SPEED) + K1*INTEGRAL + K2*SOLLSPEED$$

3. P-Lagezustandsregler ohne Stromrückführung

$$IQSOLL = K2*(SOLLAGE - LAGE) + K1*SPEED$$

4. P-Lagezustandsregler mit Stromrückführung

$$IQSOLL = K2*(SOLLAGE - LAGE) + K1*SPEED + K0*IQ$$

5. PI-Lagezustandsregler

$$INTEGRAL := INTEGRAL + SOLLAGE - LAGE$$

$$IQSOLL = K2*(SOLLAGE - LAGE) + K1*SPEED + K0*IQ + K3*INTEGRAL$$

6. Kaskadierter P-Lageregler

Dem Lageregler mit dem Reglergesetz:

$$SOLLSPEED := K3*(SOLLAGE - LAGE)$$

ist hier der unter 1. beschriebene P-Drehzahlregler unterlagert.

7. Kaskadierter PI-Lageregler

In Unterschied zum unter 6. beschriebenen Regler wird hier der unter 2. beschriebene PI-Drehzahlregler verwendet.

8. Stromreglertest

Hier ist kein übergeordneter Regler eingeschaltet.

Bei den Reglern mit I-Anteil (2,5,7) ist jeweils eine Anti-Windup-Funktion implementiert. Diese bewirkt ein Nullsetzen des Integrals, wenn die Sollwertabweichung einen vorgebbaren Wert überschreitet.

Bei den Lage-Zustandsreglern (4,5) werden Lagesollwert und Lageistwert mit den gleichen Faktoren multipliziert. Lageinvariante Zustandsregler mit ungleichen Faktoren wurden wegen der

bei der Berechnung im Festkommaformat auftretenden Schwierigkeiten nicht implementiert.

8.5.4 STROMREGLERROUTINEN

Im folgenden wird für die implementierten Algorithmen jeweils das Reglergesetz angegeben. Die Stromreglerkonstanten sind dabei mit G0 bis G3 bezeichnet.

1. BGM-STEUERUNG

$$UDSOLL = 0$$

$$UQSOLL = UZKFAKT * IQSOLL$$

2. Stromsteuerung

$$UDSOLL = UZKFAKT * (IDSOLL - SPEED * LKONST * QQ)$$

$$UQSOLL = UZKFAKT * (IQSOLL + SPEED * (LKONST * DD + EKONST))$$

Für QQ und DD werden dabei abhängig von der gewählten Konfiguration die Sollwerte IDSOLL, IQSOLL oder die Istwerte ID, IQ verwendet.

Bei Verwendung der Lageregler 4 und 5 entstehen bei Kompensation mit Istwerten einfache Lageregler ohne unterlagerte Stromregelung.

2. P-Stromregelung mit Kompensation

$$UDSOLL = UZKFAKT * (IDSOLL - SPEED * LKONST * QQ) + G0 * (IDSOLL - ID)$$

$$UQSOLL = UZKFAKT * (IQSOLL + SPEED * (LKONST * DD + EKONST)) + G0 * (IQSOLL - IQ)$$

Die Kompensation kann dabei wieder mit den Sollwerten oder den Istwerten durchgeführt werden.

3. Rotorfester Stromzustandsregler mit Totzeitberücksichtigung

UD_KOMP = UZKFAKT*(-SPEED*LKONST*IQ)

UQ_KOMP = UZKFAKT*SPEED*(LKONST*ID+EKONST)

UDSOLLO := G1*IDSOLL - G2*ID + G3*UDSOLLO

UQSOLLO := G1*IQSOLL - G2*IQ + G3*UQSOLLO

UDSOLL := UDSOLLO + UD_KOMP

UQSOLL := UQSOLLO + UQ_KOMP

4. Statorfester Stromzustandsregler mit Totzeitberücksichtigung
Aus den Werten IDSOLL und IQSOLL werden mit der Routine XYDQ die Werte IXSOLL und IYSOLL berechnet.
Aus SPEED werden mit der Routine POLRE die Polradspannungen UPX und UPY berechnet:

UXSOLLO := G1*IXSOLL - G2*IX + G3*UXSOLLO

UYSOLLO := G1*IYSOLL - G2*IY + G3*UYSOLLO

UXSOLL := UXSOLLO + UPX

UYSOLL := UYSOLLO + UPY

8.6 DIE KOMMUNIKATIONSEBENE

8.6.1 DIE STRUKTUR DER KOMMUNIKATIONSEBENE

Das Hauptprogramm besteht aus einer Initialisierungssequenz und einer nachfolgenden Scheife, in der die Befehlsinterpretation und -abarbeitung erfolgt.

Die Initialisierung enthält folgendes Abschnitte:

- Initialisierung des Prozessors
- Initialisierung der seriellen Schnittstelle
- Initialisierung des Terminals VT100 und Ausgabe des Programmtitels

- Durchführung des Nullabgleichs der Stromzähler
- Einsprung in die Interpreterschleife

Die Interpreterschleife arbeitet mit einem außerhalb des Registerfiles lokalisierten Eingabepuffer mit der Adresse REC_BUFFER. Sie verwaltet den Schreib- und Lesezeiger REC_COUNT. Die ein vereinfachtes Flußdiagramm des Hauptprogramms ist in Bild 8.4 dargestellt.

8.6.2 BEFEHLSEINGABE

Nachdem die Routine REC_BYTE ein Byte vom Terminal empfangen hat, wird es in den Eingabepuffer geschrieben und mit der Routine TRANS_BYTE zum Terminal zurückgeschickt, damit der Benutzer die eingebene Zeichenkette auch auf dem Bildschirm sieht.

Sonderfunktionen sind den Zeichen "<" und ">" zugeordnet, die nicht in den Eingabepuffer aufgenommen werden.

Wird das Zeichen "<" empfangen, so wird der Zeiger REC_COUNT auf den Anfang des Puffers gesetzt. Die Eingabe des Zeichens "<" kann also dazu verwendet werden, die bisherige Eingabe für ungültig zu erklären.

Wird das Zeichen ">" empfangen, so erfolgt nach Abarbeitung der Routine MESS_OUT die Interpretation der Zeichen im Puffer durch die Routine COM_INTERPR.

8.6.3 DIE ROUTINE MESS_OUT

Sie dient zur Ausgabe von Fehlermeldungen aus den Routinen des Ebenen 2 und 3. Die Übergabe der Fehlermeldungen erfolgt mit der Byte-Variable MESSAGE_FLAG. Einzelne Bits dieser Variablen werden von den Routinen des Ebenen 2 und 3 beim Auftreten von Fehlerbedingungen gesetzt. Nach Ausgabe der Fehlermeldung wird das betreffende Bit von der Routine MESS_OUT zurückgesetzt.

8.6.4 DER BEFEHLSINTERPRETER COM_INTERPR

Eine gültige Eingabe besteht aus einem ersten Zeichen, das die Befehlskennzeichnung enthält, und einer nachfolgenden Folge von Ziffern im HEX-Format. Zulässige Befehlskennzeichnungen sind die Buchstaben "A" bis "Z", sowie einige Zeichen, deren ASCII-Code auf den des Zeichens "Z" folgt.

Der Verzicht auf die Prüfung der Anzahl der eingegebenen Zeichen ermöglicht die Befehlswiederholung durch Eingabe des Zeichens ">".

Nach Prüfung der Befehlskennzeichnung auf Gültigkeit erfolgt mit einer Sprungtabelle der Sprung zu der Anfangsadresse der gewählten Routine.

Wird eine unzulässige Befehlskennzeichnung erkannt oder ein Fehler bei der Interpretation der nachfolgenden Zeichenkette festgestellt, so erfolgt nach Ausgabe einer entsprechenden Fehlermeldung ein Rücksprung zur Eingaberoutine.

Anderfalls wird die Befehlsroutine abgearbeitet und es erfolgt nach Ausgabe der Meldung "OK" ein Rücksprung zur Eingaberoutine. Die ausgegebenen Fehlermeldungen haben dabei folgende Bedeutung:

"SE" : unzulässige Befehlskennzeichnung
"HE" : unzulässige Zeichen in den HEX-Daten
"PE" : unzulässiger Parameterwert

8.6.5 DIE SOLLWERTVERWALTUNG

Das System verwaltet bei den Sollwerten für Strom, Drehzahl und Lage jeweils drei Variablen.

Diese seien hier mit SOLLWERT, SOLLWERT1 und SOLLWERT2 bezeichnet.

Bei der Eingabe eines Sollwerts wird der Inhalt von SOLLWERT1 der Variablen SOLLWERT2 zugeordnet. Der neueingebene Wert wird gleichzeitig den Variablen SOLLWERT und SOLLWERT1 zugewiesen. Bei den Befehlen "Soll.... wechseln" wird gesteuert durch die Byte-Variable AKT_SOLLW_FLAG einer der Werte SOLLWERT1 oder SOLLWERT2 der Variablen SOLLWERT zugeordnet. Anschließend wird die Variable AKT_SOLLW_FLAG incrementiert.

8.6.6 BEFEHLSÜBERSICHT

Im folgenden wird eine Übersicht über die implementierten Befehle gegeben.

A: Soll-Lage wechseln

B: Eingabe der Soll-Lage (8 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Umdrehungen, Rotorwinkel

Aus der Eingabe wird die Sollage im Format des Lagezählers berechnet und der Variablen SOLLAGE zugewiesen.

C: Solldrehzahl wechseln

D: Eingabe der Solldrehzahl (8 Zeichen)

Die Eingabe erfolgt als LONGINTEGER-Wert in der Reihenfolge SOLLSPPEED, SOLLSPPEED_L.

Dabei werden die unteren 16 Bit, die in der Variablen SOLLSPPEED_L stehen, nur beim Integralteil des PI-Drehzahlreglers ausgewertet. Der PI-Drehzahlregler gestattet so auch die Vorgabe sehr kleiner Drehzahlsollwerte.

E: Sollstromwerte wechseln

F: Eingabe der Stromsollwerte (8 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Sollwert ID, Sollwert IQ

G: Konstantensatz umladen (2 Zeichen)

Das erste Zeichen gibt an, ob der Konstantensatz geladen (0) oder gespeichert (1) werden soll. Die Nummer des Satzes wird durch das zweite Zeichen bestimmt.

H: Befehl HILFE

Dieser Befehl gibt eine Übersicht über die zulässigen Befehle auf dem Terminal aus.

I: Systemkonstanten und Sollwerte zeigen

Unter Bezeichnungszeilen werden die Systemkonstanten und Sollwerte gezeigt.

J: Befehl Systeminformation

Dieser Befehl gibt eine Übersicht über die gewählten Programmparameter auf dem Terminal aus.

- K: Speichern und Darstellung eines Datensatzes (1 Zeichen)
Der im eingegebenen Zeichen ("0" bis "5") beschriebene Datensatz wird in den Speicherbereich geschrieben und anschließend zusammen mit der entsprechenden Bezeichnungszeile auf dem Bildschirm ausgegeben.
- L: Wahl des übergeordneten Reglers (1 Zeichen)
Die eingegebene Reglernummer wird in der Variablen REGART gespeichert. Die Adresse des gewählten Reglers wird einer Tabelle entnommen und der Variablen REG_ADR zugewiesen.
- M: Wahl des Stromreglers (1 Zeichen)
Die eingegebene Reglernummer wird in der Variablen SREGART gespeichert. Die Adresse des gewählten Reglers wird einer Tabelle entnommen und der Variablen SREG_ADR zugewiesen.
- N: Wahl der Systemkonfiguration (4 Zeichen)
Dieser Befehl ändert die Byte-Variable CONFIG_FLAG
Gesetzte/nicht gesetzte Bits dieser Variablen haben dabei folgende Bedeutung:
Bit 0: Spannungszeiger-Nachführmechanismus ein-/ausgeschaltet
Bit 1: Störgrößenkompensation mit Ist-/Sollwerten
- O: Löschen von Maximal- und Integralwerten
Die Werte werden durch Nullsetzen neu initialisiert.
- P: Tabelle anlegen (1 Zeichen)
Das eingegebene Zeichen wird gewandelt und der Variablen TAB_ART zugewiesen. Gesteuert durch die Variablen TAB_LEN, TAB_COUNT, TRIG_ART, TRIG_WERT1 und TRIG_WERT2 wird dann eine Tabelle angelegt. Für die Tabelle steht der Speicherbereich vom Programmende bis zur Adresse 5FFFH zur Verfügung. Der Wert TAB_LEN wird gegebenenfalls durch die Routine TAB_LEN_CORR soweit herabgesetzt, daß die Tabelle in diesen Speicherbereich hineinpasst. Nachdem die Tabelle angelegt wurde, erfolgt ein Sprung zur nachfolgend beschriebenen Routine.
- Q: Tabelle zeigen
Unter einer entsprechenden Bezeichnungszeile wird die Tabelle ausgegeben.
Dabei ist eine Pagingfunktion implementiert.
Es besteht die Möglichkeit zum Blättern in der Tabelle (Eingaben: H und D) und zum vorzeitigen Ausstieg aus der Routine (Eingabe: Q).

R: Spannungsdrehfeld definieren (6 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Betrag des Spannungszeigers (2 Zeichen) und Drehgeschwindigkeit (4 Zeichen). Die eingegebenen Werte werden direkt den Variablen UBET und USPEED der Ebene 3 zugewiesen. Der Aufruf dieses Befehl ist also nur bei ausgeschalteter Regelung wirksam.

S: Spannungszeiger definieren (6 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Betrag (2 Zeichen) und Winkel (4 Zeichen) des Spannungszeigers. Aus dem gleichen Grund wie oben bleibt dieser Befehl bei eingeschalteter Regelung unwirksam.

T: Nullage setzen

Durch Eingabe dieses Befehls wird die gegenwärtige Rotorlage zur Nullage deklariert. Dies erfolgt, indem der Lagezähler mit dem der Nullage zugewiesenen Zählerwert geladen wird.

U: Eingabe der Triggerzeiten (8 Zeichen)

Die Eingabe erfolgt in der Reihenfolge TRIG_WERT1 (4 Zeichen) und TRIG_WERT2 (4 Zeichen).

TRIG_WERT1 bestimmt dabei die Zeit in Reglertakten, nach der die gewählte Triggeroption (TRIG_ART) wirksam wird.

Dieser Wert wird bei der Bearbeitung des Befehls "Tabelle anlegen" benutzt.

TRIG_WERT2 legt die Zeit in Reglertakten fest, nach der entsprechend der gewählten Triggeroption ein Sollwertwechsel durchgeführt wird. Dieser Wert wird von den Routinen zum Sollwertwechsel benutzt.

V: Werte im Konstantenarray ändern (6 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Zeile (1 Zeichen), Spalte (1 Zeichen) und Wert (4 Zeichen) der Konstanten.

W: Systemkonstanten ändern (5 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Konstantennummer (1 Zeichen) und Wert (4 Zeichen).

X: Konstantenarray zeigen

Y: Sollwertwechselroutine wählen (9 Zeichen)

Der eingegebene Wert wird in der Variablen CNGART gespeichert. Die Adresse der gewählten Routine wird einer Tabelle entnommen und der Variablen CNG_ADR zugewiesen.

Z: Tabellenparameter festlegen (9 Zeichen)

Die Eingabe erfolgt in der Reihenfolge Tabellenlänge
(TAB_LEN: 4 Zeichen) zeitlicher Abstand der Tabelleneinträge
(TAB_FAKTOR: 4 Zeichen) und Tiggerart (TRIG_ART: 1 Zeichen).

[: Totzeitstromreglerkonstanten berechnen

Ausgehend von einer Konstanten werden die beiden anderen
Konstanten des Totzeitstromregler berechnet.

\ : Nicht implementiert

] : Interrupt ein-/ausschalten (1 Zeichen)

Das Sperren des Interrupts bewirkt, daß nur noch der Komman-
dointerpreter aktiv ist.

^ : Reglertaktzeit festlegen (1 Zeichen)

Eingegeben wird hier die Zahl der Interrupte pro Reglertakt.
Dabei ist minimal der Wert 4 und maximal der Wert 7 zulässig.
Da der Interruptabstand 258 ms beträgt, lassen sich somit
Reglertaktzeiten im Bereich von 1,024 ms bis 1,792 ms ein-
stellen.

- : Test der Transformationsroutinen (max 13 Zeichen)

Der Test erfolgt bei ausgeschaltetem Interrupt. Die Eingangs-
werte für die gewählte Routine werden eingegeben.
Anschließend erfolgt die Anzeige der Ausgangswerte. Getestet
werden können die Routinen POLRE, REPOL und XYDQ.
Da dieser Programmteil in dem Speicherbereich angesiedelt
ist, der der Tabelle zugewiesen ist, erfolgt die Ausführung
dieses Befehls nur dann, wenn noch keine Tabelle angelegt
wurde.

8.7 BESCHREIBUNG EINZELNER RECHENROUTINEN

8.7.1 DIE ROUTINE REPOL

Die Routine REPOL führt die Transformation vom karthesischen
Koordinatensystem ins Polarkoordinatensystem durch. Eingangs-
größen sind die Variablen XX und YY. Ausgangsgrößen sind die
Variablen BET und PHA. Mathematisch sind die Transformationsglei-

chungen gegeben durch:

$$BET = \sqrt{XX^2 + YY^2}$$

$$PHA = \arg(XX + j*YY)$$

Besondere Probleme bereitet hier die Wurzelfunktion. Da die Quadratwurzelfunktion stark nichtlinear ist und ein großer Eingangswertebereich auftritt, kann die Wurzelziehung mit tabellierter Wurzelfunktion nicht im Festkommaformat erfolgen.

Statt dessen wird der Radikand zunächst in ein Gleitkommaformat umgewandelt, das die Ausnutzung der Beziehung:

$$\sqrt{x} = \sqrt{x*2^{2n}} / 2^n$$

gestattet. Der Exponent n wird dabei so groß gewählt, daß der Wert $x*2^{2n}$ gerade noch in das LONGWORD-Datenformat passt.

Die Multiplikation mit und die Division durch eine Zweierpotenz wird dabei durch Links- und Rechtsschieben des Longwords realisiert. Bei der Multiplikation mit 2^{2n} wird der Prozessorbefehl NORML benutzt. Entsteht dabei eine ungerade Shiftzahl, so wird der Operand anschließend um eine Dualstelle nach rechts verschoben.

Zur eigentlichen Wurzelbildung wird dann nur das oberste Byte des Ausdrucks $z = x*2^{2n}$ verwendet. Ist y das Addressoffsetbyte, so liefert die Wurzeltabelle den Wert $y*2^3$ im Byteformat. Die durch das Byteformat vorgegebene Genauigkeit wird somit voll ausgeschöpft. Da der Fall $x=0$ separat behandelt wird, reicht eine Wurzeltabelle der Länge $(255 - 63)\text{Byte} = 192\text{ Byte}$ aus.

Die maximale Differenz $z - y*2^{2n}$ beträgt $2^{2n}-1$. Unter Berücksichtigung der Ungleichung $z \geq 2^{2n}$ läßt sich der maximale Eingangsfehler der Wurzeltabelle abschätzen zu 2^8 .

Der maximale relative Eingangsfehler bei Rundung des Arguments beträgt demzufolge $2^7/2^{15} = 2^{-8}$. Wegen des verwendeten Gleitkommaformats liefert die Wurzeltabelle einen nahezu konstanten maximalen relativen Ausgangsfehler. Dieser beträgt $2^5/2^{15} = 2^{-10}$.

Der maximale relative Fehler der so durchgeführten Wurzelziehung beträgt somit ca $2^{-8} = 0,4\%$. Da die beschriebene Routine im

Programm zur Transformation des Spannungszeigers benutzt wird, dessen Betrag maximal 128 betragen kann, entsteht somit ein maximaler absoluter Fehler von ± 1 .

Auch bei der Berechnung des Winkels PHA unter Benutzung der Arcus-Tangens-Funktion sind besondere Überlegungen notwendig. Zunächst wird anhand der Vorzeichen von XX und YY der Quadrant bestimmt, in dem der Winkel PHA liegt.

Die Berechnung des Hilfswinkels β erfolgt dann gemäß:

$$\beta = \arctan \frac{|YY|}{|XX|} \quad \text{für } |YY| < |XX|$$

$$\beta = 45^\circ \quad \text{für } |YY| = |XX|$$

$$\beta = 90^\circ - \arctan \frac{|XX|}{|YY|} \quad \text{für } |YY| > |XX|$$

Die Division erfolgt so, daß der Dividend in das HIGH-WORD des Dividendenregisters geladen wird, während das LOW-WORD zu Null gesetzt wird. Die hier gewählte Vorgehensweise stellt sicher, daß bei der Division kein Überlauf stattfindet. Das Resultatword der Division enthält den Quotienten $\cdot 2^{16}$. Das obere Byte des Resultatwords wird als Addressoffset für den Zugriff auf die Arcus-Tangens-Tabelle benutzt. Benötigt wird also eine Tabelle mit einer Länge von 256 Byte. Nachdem der Hilfswinkel β bestimmt worden ist, kann unter Berücksichtigung des Quadranten der Winkel PHA bestimmt werden:

$$\begin{array}{ll} x > 0, y > 0 : & \text{PHA} = \beta \\ x < 0, y > 0 : & \text{PHA} = 180^\circ - \beta \\ x > 0, y < 0 : & \text{PHA} = 180^\circ + \beta \\ x < 0, y < 0 : & \text{PHA} = 360^\circ - \beta \end{array}$$

Der hier allein interessierende maximale absolute Fehler $\Delta\beta$ wird im wesentlichen durch den Eingangsfehler der Arc-Tan-Tabelle verursacht. Er läßt sich nach folgender Formel bestimmen:

$$\Delta\beta = \arctan (1/256) = 0,22^\circ$$

Dieser Fehler ist durchaus tolerierbar.

8.7.2 DIE ROUTINE XYDQ

Die Routine XYDQ führt die Umrechnung vom statorfesten ins rotor-
feste Koordinatensystem durch. Eingangsgrößen sind die Variablen
XX, YY und PHA. Ausgangsgrößen sind die Variablen DD und QQ.
Mathematisch sind die Transformationsgleichungen gegeben durch:

$$DD = XX \cdot \cos(PHA) + YY \cdot \sin(PHA)$$

$$QQ = -XX \cdot \sin(PHA) + YY \cdot \cos(PHA)$$

Die auftretenden Winkelfunktionswerte lassen sich aus einer
Sinustabelle für den Bereich $0 \leq \beta \leq 90^\circ$ bestimmen. Die Sinus-
tabelle liefert dabei den Wert $\sin(\beta) \cdot 128$ im Byte-Format.
Durch die Wahl des Faktors 128 anstelle von 256 ist sicherge-
stellt, daß auch $\sin(90^\circ)$ noch korrekt dargestellt werden kann.
Bei der Berechnung eines Funktionswerts muß aus dem Winkel $\varepsilon =$
PHA der Hilfswinkel β berechnet werden. Zugleich muß das Vorzei-
chen VZ des Funktionswerts bestimmt werden.

Für den Sinus gilt:

	$0^\circ < \varepsilon < 90^\circ$	$90^\circ < \varepsilon < 180^\circ$	$180^\circ < \varepsilon < 270^\circ$	$270^\circ < \varepsilon < 360^\circ$
β	ε	$180^\circ - \varepsilon$	$\varepsilon - 180^\circ$	$360^\circ - \varepsilon$
VZ	+	+	-	-

Für den Cosinus gilt:

	$0^\circ < \varepsilon < 90^\circ$	$90^\circ < \varepsilon < 180^\circ$	$180^\circ < \varepsilon < 270^\circ$	$270^\circ < \varepsilon < 360^\circ$
β	$90^\circ - \varepsilon$	$\varepsilon - 90^\circ$	$270^\circ - \varepsilon$	$\varepsilon - 270^\circ$
VZ	+	-	-	+

Der bei der Tabellierung entstehende absolute Fehler beträgt
maximal $0,5 / 128 = 0,004$.

Die Genauigkeit der Berechnung wird wesentlich bestimmt durch die
gewählte Winkelauflösung. Bei einer Aufteilung des rechten
Winkels in 256 Teile erhält man eine Winkelauflösung von $\beta_0 =$
 $0,35^\circ = 0,006 \text{ rad}$.

Der maximale absolute Fehler des Sinus bzw Cosinus aufgrund der
begrenzten Winkelauflösung beträgt dann:

$$\sin \beta_0 = 0,006$$

Hieraus errechnet sich der maximale absolute Fehler ΔDD bzw ΔQQ zu:

$$\Delta DD \leq (|XX| + |YY|) * (0,006 + 0,004) < \sqrt{|XX|^2 + |YY|^2} * 0,02$$

Dieser Fehler von maximal 2 % der Zeigerlänge ist akzeptabel.

8.7.3 DIE ROUTINE POLRE

Die Routine POLRE führt die Umrechnung vom Polarkoordinatensystem ins karthesische Koordinatensystem durch. Eingangsgrößen sind die Variablen BET und PHA. Ausgangsgrößen sind die Variablen XX und YY.

Mathematisch sind die Transformationsgleichungen gegeben durch:

$$XX = BET * \cos(PHA) \qquad YY = BET * \sin(PHA)$$

Die auftretenden Winkelfunktionswerte werden wieder aus der oben beschriebenen Sinustabelle bestimmt.

8.7.4 LAGEAUSWERTUNG

Aus dem vom Lagezähler gelieferten Lagewert werden durch Division durch die Zahl 3600 der Rotorwinkel INCREM und die Anzahl der el. Umdrehungen EL_UMDR berechnet.

Aus der Bedingung (8.3.4.1) folgt nun, daß für die Anzahl der el. Umdrehungen nur ein Wertebereich von 2^{16} zulässig ist. Dieser Wertebereich ist für das hier beschriebene Versuchsprogramm ausreichend. Für eine industrielle Applikation ist jedoch ein größerer Wertebereich erforderlich. Die erhebliche Vergrößerung des Wertebereichs kann dadurch erfolgen, daß zunächst eine vorzeichenlose Division des Lagewerts durch $3600 * 18$ erfolgt. Der Rest der Division wird anschließend durch 3600 dividiert:

Dies führt zu folgender Programmsequenz:

```
a1 := NEULAGE div (3600*18);  
a2 := NEULAGE mod (3600*18);  
b1 := a2 div 3600;  
INCREM := a2 mod 3600;  
EL_UMDR := a1*18 + b1;
```

Hierdurch vergrößert sich der Wertebereich um den Faktor 18.

Ein beliebig erweiterbarer Wertebereich für die Lage läßt sich dadurch erzielen, daß Rotorwinkel und Zahl der Umdrehungen nicht durch Division, sondern durch ständige Aktualisierung gewonnen werden. Hierzu ist folgende Programmsequenz erforderlich:

```
INCREM := NEULAGE - HILFSLAGE;  
if INCREM >= 3600 then  
begin  
  INCREM := INCREM - 3600;  
  HILFSLAGE := HILFSLAGE + 3600;  
  EL_UMDR := EL_UMDR + 1;  
end else  
if INCREM < 0 then  
begin  
  INCREM := INCREM + 3600;  
  HILFSLAGE := HILFSLAGE - 3600;  
  EL_UMDR := EL_UMDR - 1;  
end;
```

Um die Länge der benutzten trigonometrischen Tabellen klein zu halten, ist es nun sinnvoll, die durch den Incrementalgeber bestimmte Winkelauflösung zu reduzieren. Die Reduzierung hat dabei nur Einfluß auf die Genauigkeit der Transformationen von Strom- und Spannungszeigern, nicht jedoch auf die Lageregelung, da dort der Istwert NEULAGE benutzt wird.

Die Reduzierung ist zulässig, da einerseits der mittlere Winkelfehler des Strommesswerts i bereits wesentlich größer ist als die gegebene Winkelauflösung und andererseits der durch das Verfahren bestimmte mittlere Winkelfehler bei der Realisierung der Stellgröße wesentlich größer ist als die gegebene Winkelauflösung.

Angestrebt wird eine Reduzierung um den Faktor 4. Die reduzierte Winkelauflösung beträgt dann 0,4 Grad.

Der durch Umrechnung gewonnene reduzierte Winkelwert wird mit

INCEM_INT bezeichnet.

Im Programm sind zwei verschiedene Möglichkeiten zur Reduzierung der Winkelauflösung vorgesehen, die wahlweise realisiert werden können:

a) 900 Incremente pro el. Umdrehung:

INCREM_INT := INCREM div 4

b) 1024 Incremente pro el. Umdrehung:

INCREM_INT := trunc (INCREM * (64/225))

Nachteilig bei Methode a) ist, daß bei Additionen und Subtraktionen von Winkeln aufwendige Modularechnungen durchgeführt werden müssen. Die Division durch 4 kann hingegen durch Rechtsschieben des Operanden INCREM erfolgen.

Bei Methode b) kann die Modularechnung für den Winkel WINKEL mit dem Befehl

AND WINKEL, #1023D

erfolgen. Zur Berechnung von INCREM_INT ist hier eine Multiplikation notwendig.

Methode b) bietet Vorteile, da Modularechnungen mehrfach im Programmablauf erfolgen. Zudem kann die Anpassung an einen Incrementalgeber mit anderer Winkelauflösung bei Verwendung von Methode b) durch Änderung des Multiplikationsfaktors erfolgen, während bei Verwendung von Methode a) die im Programm benutzten trigonometrischen Tabellen geändert werden müssen.

8.7.5 Drehzahlberechnung

Bei der Drehzahlberechnung wird der Zähler SPEED_COUNT benutzt. Er wird bei jedem Interrupt incrementiert. Die Drehzahlberechnung wird nur dann durchgeführt, wenn SPEED_COUNT größer oder gleich der Konstanten SPEED_COUNT_MIN ist. Diese Konstante legt somit die minimale Messdauer für die Drehzahlberechnung fest. Durch Wahl eines größeren Wertes SPEED_COUNT_MIN kann der absolute Fehler des Drehzahlwertes verringert werden. Diese Verringerung geht allerdings auf Kosten der Aktualität des Drehzahlwerts.

Die Berechnung der Drehzahl SPEED erfolgt nach der Formel:

SPEED := (NEULAGE - ATTLAGE) * 64/SPEED_COUNT

Bei der Bildung der Differenz reicht es aus, die LOW-Worte der

Operanden NEULAGE und ALTLAGE zu subtrahieren, wenn die Differenz NEULAGE - ALTLAGE in das WORD-Format paßt. Dies ist sichergestellt, wenn SPEED_COUNT_MIN $\leq$ 64 ist.

Der Faktor 64/SPEED_COUNT wird einer Tabelle entnommen. Der Wert SPEED gibt also die Zahl der Incremente pro 16 ms an. Nach erfolgter Drehzahlberechnung wird SPEED_COUNT auf den Wert 0 gesetzt.

8.7.6 DIE BEGRENZERROUTINE LIMIT

Die Begrenzeroutine dient dazu, einen Longintegerwert auf eine wählbare Dualstellenzahl zu begrenzen.

Sie übernimmt den zu begrenzenden LONGINTEGER-Wert im Registerpaar B1X,C1X .

In der Routine wird ein Maskenlongword MASK,MASKH verwendet. Bei diesem Maskenlongword, das abhängig von der gewählten Stellenzahl einer Tabelle entnommen wird, sind alle zu zulässigen Dualstellen gehörenden Bit zu 0 gesetzt, während alle anderen Bits zu 1 gesetzt sind.

Zunächst erfolgt eine Undierung des Longwords B1X,C1X mit dem Maskenlongword MASK,MASKH.

Das weitere Vorgehen ist abhängig vom Vorzeichen des Eingangswerts. Bei positiven Werten ist eine Begrenzung nicht notwendig, wenn das Ergebnis der Undierung 0 ist. Bei negativen Werten ist eine Begrenzung nicht notwendig, wenn das Ergebnis der Undierung gleich dem Maskenlongword ist.

Andernfalls wird das Registerpaar abhängig vom Vorzeichen mit einem aus dem Maskenlongword abgeleiteten Maximalwert geladen. Die bei einem Regelalgorithmus zu wählende Dualstellenzahl für die Begrenzeroutine ergibt sich aus der Anzahl der durchzuführenden Additionen und Subtraktionen.

8.3.7.7 Kompensationsberechnungen

Die zum stationären Zustand gehörende Spannungsgleichung ist in der Form:

$$UD = UZKFAKT * (ID - SPEED * LKONST * IQ * 2^{-SPNEX})$$

$$UQ = UZKFAKT * (IQ + SPEED * (LKONST * ID + EKONST) * 2^{-SPNEX})$$

implementiert.

Für $UBET = \sqrt{UD^2 + UQ^2}$ gilt :

$$UBET = 128 * \frac{|u|}{(2/3 U_{2x})}$$

Die hier gewählte Implementierung bewirkt, daß bei Änderung der Zwischenkreisspannung U_{2x} nur die Konstante UZKFAKT geändert werden muß.

Die hier auftretenden Konstanten UZKFAKT, LKONST und EKONST werden als Systemkonstanten bezeichnet.

Die Konstanten UZKFAKT und LKONST sind Gleitkommakonstanten, während EKONST eine Festkommakonstante und SPNEX ein Byte ist. Zur Vereinfachung der Darstellung wurde in Formeln der Faktor 2^{-SPNEX} weggelassen.

8.8 RECHENZEITEN

Bei der Programmversion, in der die Transformationsroutinen als MACROS realisiert wurden, konnte selbst bei der rechenintensivsten Reglerkombination (PI-Lagezustandsregler mit statorfestem, die Totzeit berücksichtigenden Stromzustandsregler) die Regel-taktzeit von 1,024 ms eingehalten werden.

8.9 HILFSPROGRAMME PARAGEN UND OBJCONV

8.9.1 DAS PROGRAMM PARAGEN

Da die Umrechnung der System- und Reglerkonstanten in die Maschinennormierung sehr aufwendig ist, wurde hierzu das Hilfsprogramm PARAGEN erstellt. Unter Maschinennormierung wird dabei die durch die Auflösung der Stromzähler, des Incrementalgebers und der Stellgröße gegebene Normierung verstanden. Das Programm rechnet die System- und Reglerkonstanten in die Maschinen-

normierung um und gibt sie in HEX-Form in dem beschriebenen Gleitkommaformat aus.

Eingangsgrößen des Programms sind die Maschinenparameter (L,R, und k_{EMK}) sowie die verwendete Reglertaktzeit, die Empfindlichkeit der Stromzähler, die Auflösung der Stellgröße und die verwendete Zwischenkreisspannung U_{2K} .

Diese Daten können in eine Datei gespeichert werden und aus dieser Datei gelesen werden.

Neben der Umrechnung von eingegebenen Reglerkonstanten ist hier auch die Berechnung der Konstanten des Totzeitstromreglers möglich.

8.9.2 DAS PROGRAMM OBJCONV

Der Befehl "SAVE" des Entwicklungssystems SBE96 ermöglicht die Abspeicherung der Daten eines wählbaren Addressbereichs in eine Datei. Hierzu sind die Anfangs- und Endadresse, der Dateiname und die Option "O" anzugeben. Die Wahl dieser Option bewirkt, daß die Daten im INTEL-OBJ-Format gespeichert werden. Die Anfangs- und Endadresse einer erstellten Wertetabelle wird vom Assemblerprogramm angezeigt.

Das Programm OBJCONV dient zur Auswertung mit dem Befehl "SAVE" erstellten Datei. Diese Datei muß das Suffix "DOB" besitzen. Das Programm OBJCONV erzeugt dabei die beiden vom Grafikprogramm GREGA benötigten Dateien.

Zur Datenkonvertierung werden die folgenden Parameter benötigt:

- Anzahl der Werte pro Datensatz
- Datentypen der Werte
- Bezeichnungen und Einheiten der Werte

Die Konvertierungsparameter können in eine Datei (Suffix CNV) gespeichert werden und aus dieser Datei gelesen werden.

Die wählbaren Datentypen sind:

BYTE, SHORTINTEGER, WORD, INTEGER, LONGWORD, LONGINTEGER.

Zusätzlich zur Konvertierung kann vom Programm auch eine Auswertung der Daten vorgenommen werden. Es sind u.a. folgende Auswertungsarten implementiert:

- Berechnung der Komponenten eines Zeigers im rotorfesten System aus den Komponenten im statorfesten System

- Berechnung der Komponenten eines Zeigers im statorfesten System aus den Komponenten im rotorfesten System
- Berechnung von Betrag und Winkel eines Zeigers
- Umrechnung der Werte aus der Maschinennormierung in Absolutwerte

Die für die Auswertung benötigten Maschinen- und Systemdaten gehören ebenfalls zu den Konvertierungsparametern.

Weiterhin ist die Eingabe von Kommentaren möglich. Diese Kommentare werden in die Parameterdatei des Grafikprogramms aufgenommen und ermöglichen die Kennzeichnung der Systemeinstellung (Wertabstand, Reglertyp, Reglerparameter) bei der Meßwertaufnahme.

9. SIMULATIONS- UND MESSERGEBNISSE

9.1 MASCHINENPARAMETER

Die Messungen wurden mit dem Motor SE 638 der Fa. MAVILOR durchgeführt. Die Daten dieses Motors wurden auch bei der Simulation verwendet. Verfügbar war außerdem der Motor SE 718 der Fa. MAVILOR. Die Kenndaten der beiden Motoren sind in Tabelle 9.1 zusammengestellt. Bei den Angaben des Herstellers /5/ sind die Spannungen jeweils zwischen zwei Maschinenklemmen gemessen worden. Die Kenndaten wurden deshalb aus den Herstellerangaben durch Umrechnung nach den in /4 HETZEL/ angegebenen Formeln gewonnen. Die Bezugsgrößen und die bezogenen Maschinenparameter der beiden betrachteten Motoren sind in Tabelle 9.2 zusammengestellt. Dabei wurde eine Zwischenkreisspannung von 250 V zugrundegelegt.

	SE 638	SE 718
Leistung P	690 W	1150 W
max. Drehzahl n_{max}	100 1/s	100 1/s
Nenndrehmoment ($n = 50 \text{ s}^{-1}$)	2,2 Nm	3,6 Nm
Impulsdrehmoment ($n = 0 \text{ s}^{-1}$)	26 Nm	35 Nm
k_{EMK}	0,41 Vs/rad	0,41 Vs/rad
Strangwiderstand R	3,4 Ω	1,4 Ω
Stranginduktivität L	5,8 mH	4,0 mH
T_{el}	1,7 ms	2,8 ms
T_{mech}	6,9 ms	3,9 ms
Rotorträgheitsmom. J	$3 \cdot 10^{-4} \text{ kgm}^2$	$4 \cdot 10^{-4} \text{ kgm}^2$

Tabelle 9.1 : Kenndaten der Motoren

	SE 638	SE 718
Bezugsdrehzahl ω_0	588 rad/s	357 rad/s
Bezugsspannung U_0	49 V	30 V
Bezugsstrom I_0	14,5 A	21,0 A
W'_{max}	4,3	7,0
u'_{max} ($U_{zk}=250$ V)	2,9	4,8
$i'_{max, D}$	0,214	0,266
$i'_{max, K}$	2,48	2,33
K_{MOM}'	7,3 Nm	10,6 Nm
J'	6,3 Nm	26,3 Nm

Tabelle 9.2 : Bezugsgrößen und bezogene Kenn-
daten der Motoren

Bild 9.1.1 zeigt das Zusammenwirken der erstellten Software zur Gewinnung von Mess- und Simulationsergebnissen.

Bei den im folgenden auftretenden Drehzahl- und Winkelangaben handelt es sich um Größen, die auf das elektrische System bezogen sind. Die zugehörigen realen mechanischen Größen erhält man durch Division durch die Polpaarzahl $z_p = 4$.

9.2 SYSTEMVERHALTEN

Bild 9.2.1 zeigt den gemessenen zeitlichen Verlauf der Stromkomponenten i_x , i_y und i_d , i_q bei Vorgabe eines Spannungsdrehfelds. Der Motor arbeitet hier als konventioneller Synchronmotor.

Bild 9.2.2 zeigt die gemessenen Stromkomponenten i_d , i_q bei Vorgabe eines Spannungsdrehfelds. Aus diesen Werten wurde der Spannungszeiger $\underline{u}$ mit der Routine MTEST zurückberechnet. Das Bild zeigt zusätzlich die zurückberechneten Werte von Betrag und Winkel des Spannungszeigers $\underline{u}$ im rotorfesten System.

9.3 STROMREGELUNG UND STROMSTEUERUNG

Zur in Abschnitt (4.2) beschriebenen Stromsteuerung wurden die folgenden Ergebnisse erzielt.

Bild 9.3.1 zeigt den gemessenen Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 0$ U/min) .

Bild 9.3.2 zeigt den gemessenen Stromverlauf bei Stromsteuerung unter Vorgabe eines Sprungs des Sollstroms.

Bild 9.3.3 zeigt den simulierten Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 350$ U/min) .

Zur in Abschnitt (4.5.2) beschriebenen Stromzustandsregelung im rotorfesten System mit Totzeitberücksichtigung wurden die folgenden Ergebnisse erzielt.

Bild 9.3.4 zeigt den gemessenen Stromverlauf bei einem Sprung des Sollstroms.

Bild 9.3.5 zeigt den gemessenen Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 0$ U/min) .

Bild 9.3.6 zeigt den simulierten Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 350$ U/min) .

Zur in Abschnitt (4.5.4) beschriebenen Stromzustandsregelung im statorfesten System mit Totzeitberücksichtigung wurden die folgenden Ergebnisse erzielt.

Bild 9.3.7 zeigt den gemessenen Stromverlauf bei einem Sprung des Sollstroms.

Bild 9.3.8 zeigt den gemessenen Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 0$ U/min).

Bild 9.3.9 zeigt den simulierten Stromverlauf unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 350$ U/min) .

Bei den Simulationen mit rechteckförmigem Sollstromverlauf zeigt die Stromsteuerung ein deutlich schlechteres Verhalten als die beiden anderen verwendeten Reglerstrukturen. Bei der Regelung im statorfesten System ist eine gegenüber der Regelung im rotorfesten System bessere Führung insbesondere der d-Komponente des Stroms erkennbar.

Im Unterschied zur Simulation war bei den Messungen mit rechteckförmigem Sollstromverlauf die Rotoranfangsdrehzahl Null. Deshalb treten die Unterschiede im Verhalten der Regler bei den Messergebnissen nicht so deutlich hervor.

9.4 LAGEREGELUNG

Bei Verwendung des kaskadierten Lagezustandsreglers wurden die folgenden Simulationsergebnisse erzielt. Als Stromregler wurden dabei der Stromregler im rotorfesten System mit Berücksichtigung der Totzeit und der Stromregler im statorfesten System mit Berücksichtigung der Totzeit eingesetzt.

Bild 9.4.1 zeigt die simulierte Systemantwort auf einen Lageführungssprung von 10° .

Bild 9.4.2 zeigt die simulierte Systemantwort auf einen Lageführungssprung von 20° .

Der zeitliche Verlauf der mechanischen Größen ist dabei unabhängig vom gewählten Stromregler. Die Führung der d-Komponente des Stroms ist beim statorfesten Stromregler geringfügig besser als beim rotorfesten Stromregler.

Bei Verwendung des kaskadierten Lagezustandsreglers mit unterlagertem Stromregler im rotorfesten System mit Berücksichtigung der Totzeit wurden die folgenden Messergebnisse erzielt. Die Zwischenkreisspannung U_{zk} betrug dabei 75 V , die Anfangsdrehzahl n_0 war 0.

Bild 9.4.3 zeigt die gemessene Systemantwort auf einen Lageführungssprung von 10° .

Bild 9.4.4 zeigt die gemessene Systemantwort auf einen Lageführungssprung von 20° .

Bild 9.4.5 zeigt die gemessene Systemantwort auf einen Lageführungssprung von 360° .

Bild 9.4.6 zeigt die gemessene Systemantwort auf einen Lageführungssprung von 8 Umdrehungen.

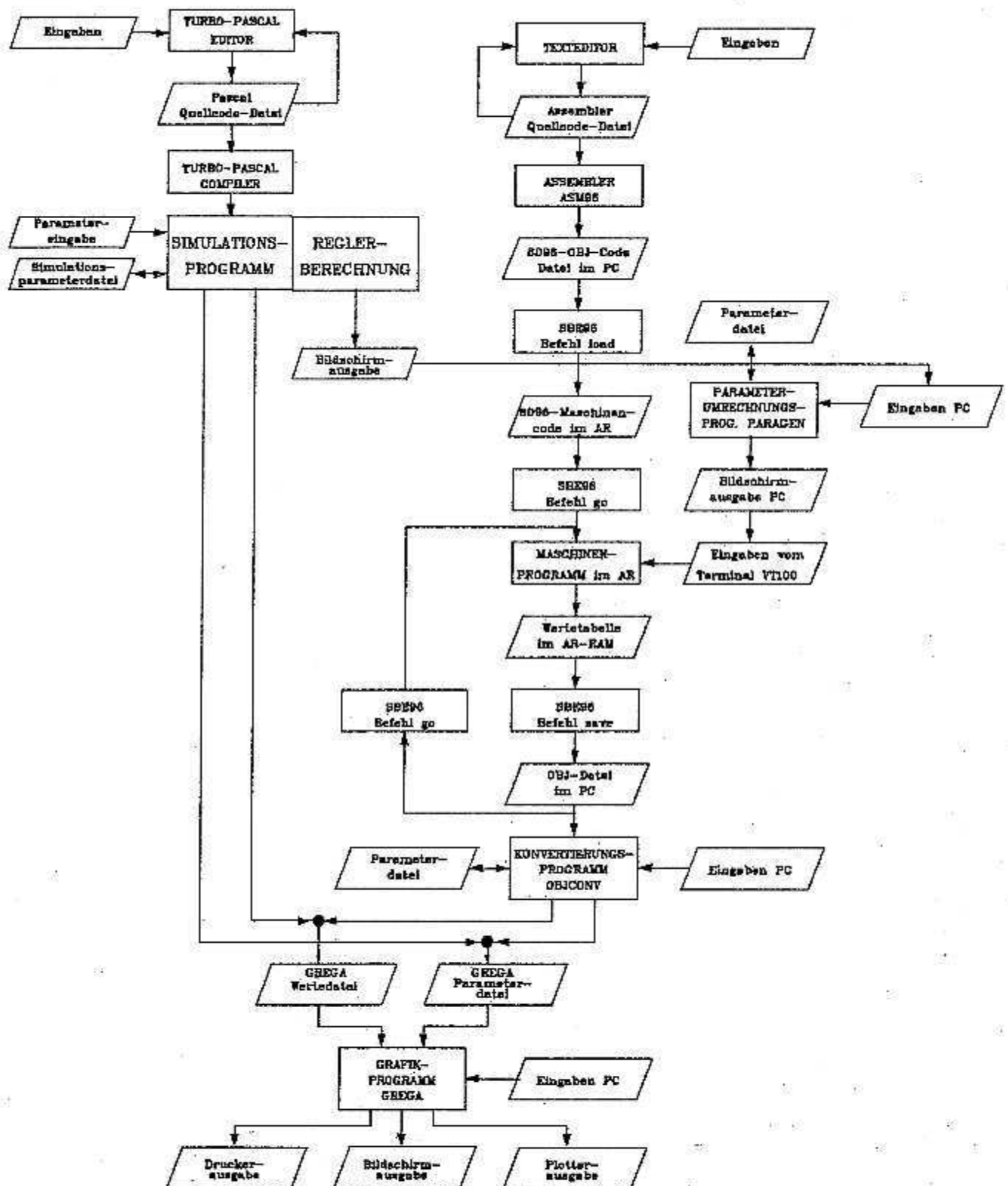


Bild 9.1.1 Zusammenwirken der erstellten Softwaremodule

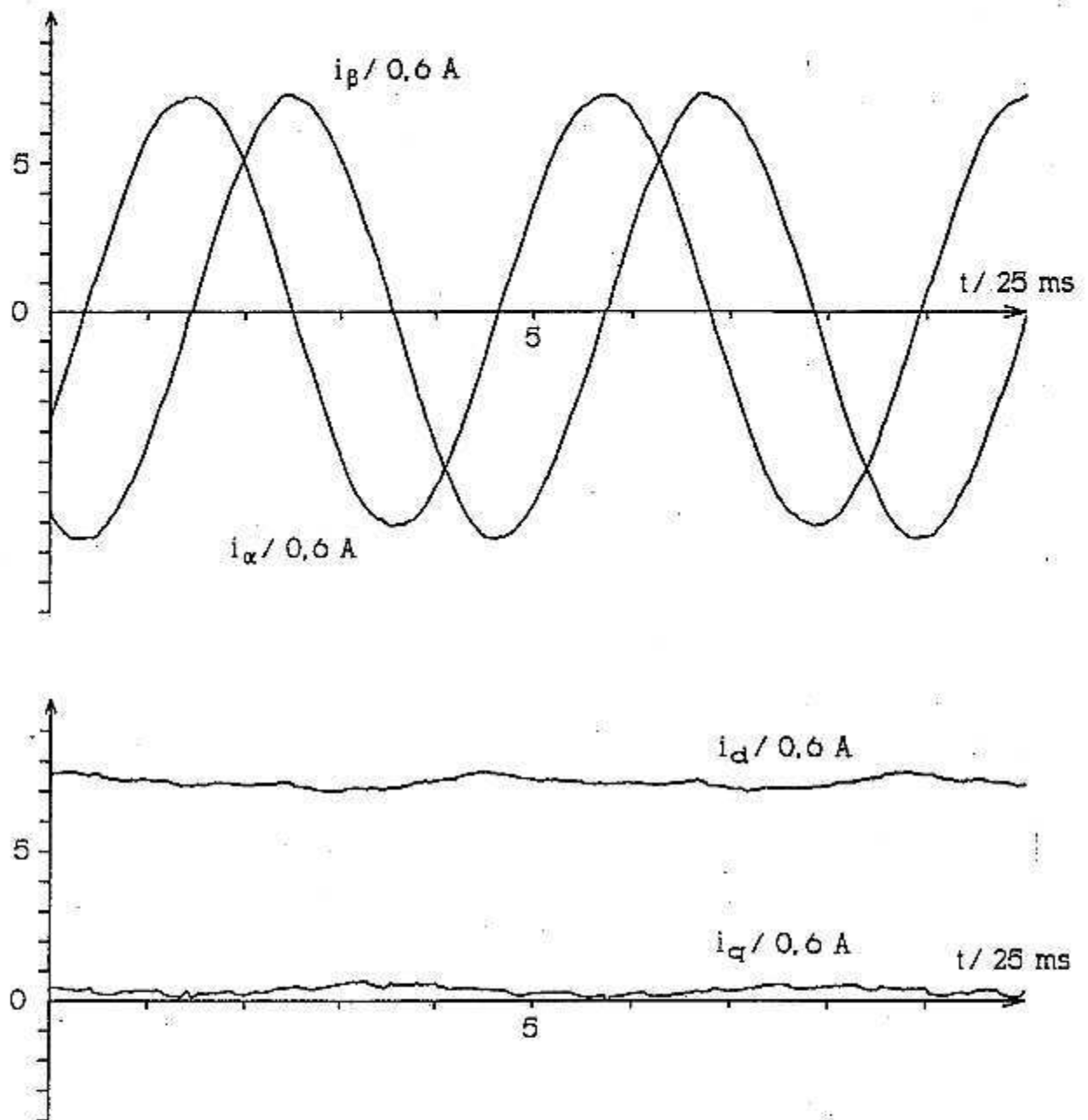


Bild 9.2.1 Gemessener zeitlicher Verlauf der Stromkomponenten i_x , i_y und i_d , i_q bei Vorgabe eines Spannungsdrehfelds

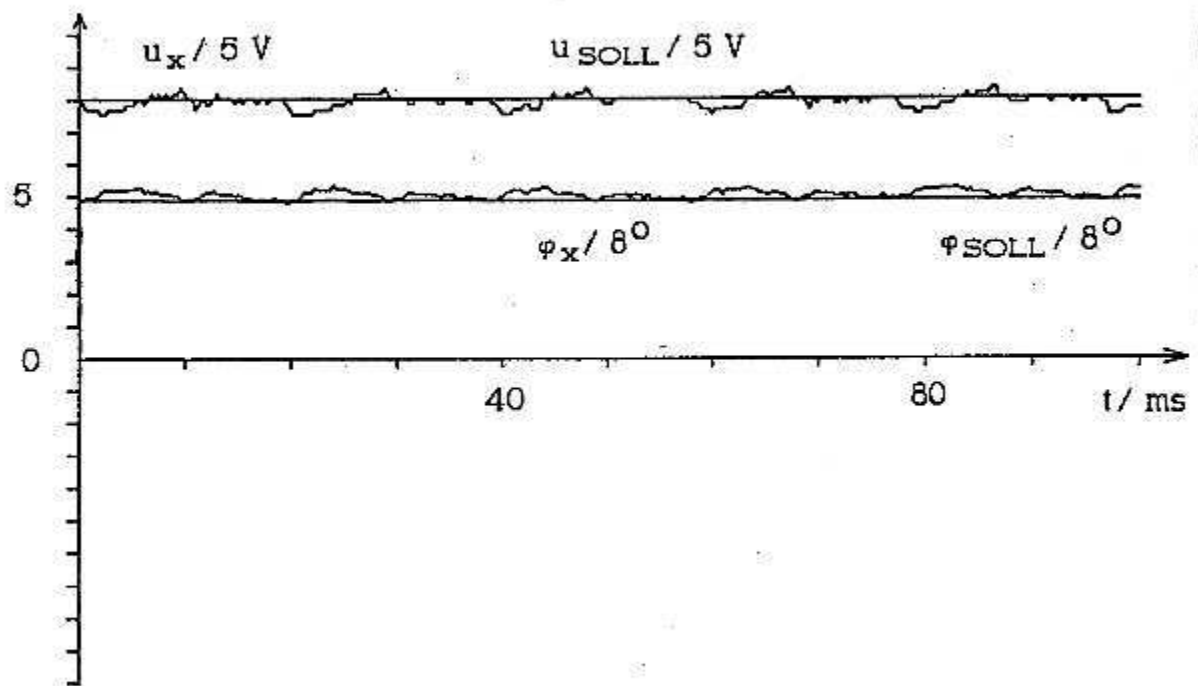


Bild 9.2.2 Zeitlicher Verlauf von Betrag und Phase des realen und des zurückberechneten Spannungszeigers $\underline{u}$ bei Vorgabe eines Spannungsdrehfelds.

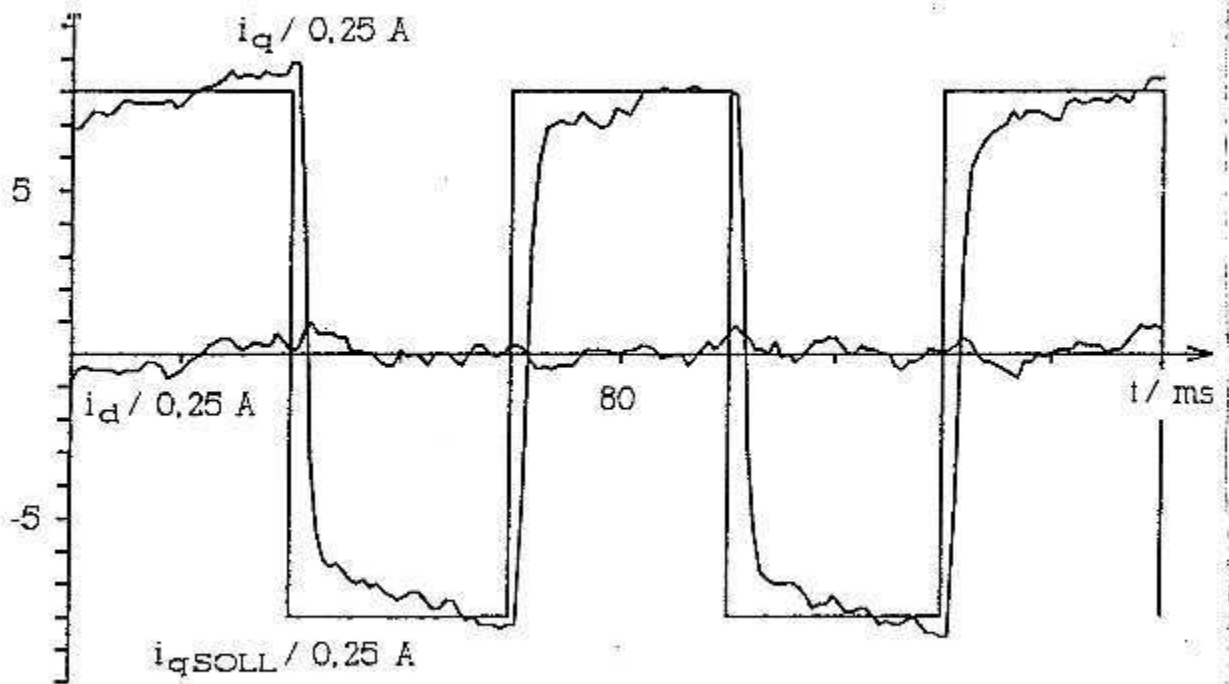


Bild 9.3.1 Gemessener Stromverlauf bei Stromsteuerung unter Vorgabe eines rechteckförmigen Sollstromverlaufs

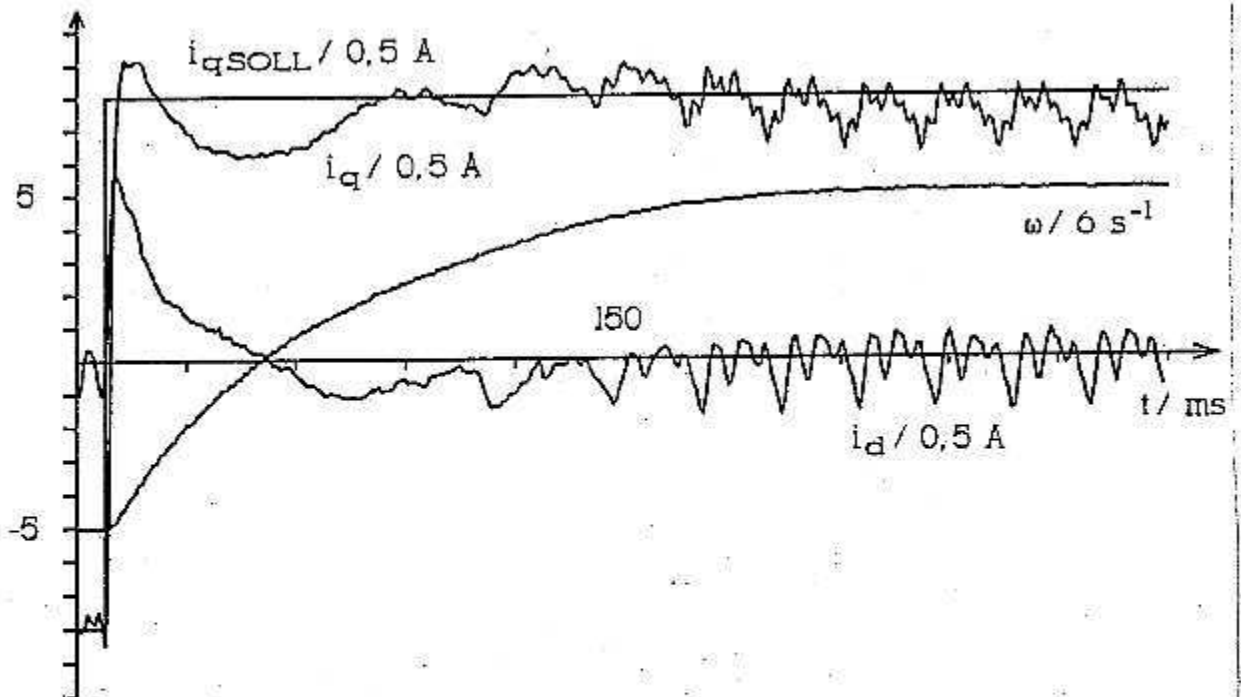


Bild 9.3.2 Gemessener Stromverlauf bei Stromsteuerung unter Vorgabe eines Sprungs des Sollstroms ($n_0 = 0 \text{ U/min}$)

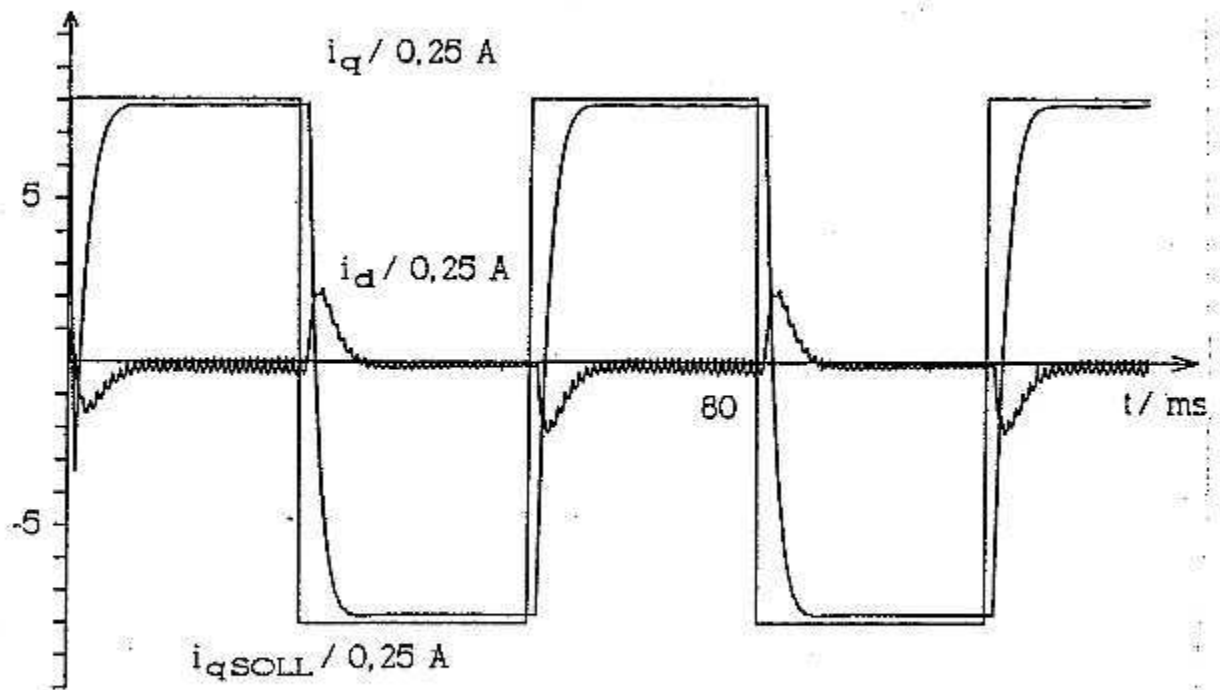


Bild 9.3.3 Simulierter Stromverlauf bei Stromsteuerung unter Vorgabe eines rechteckförmigen Sollstromverlaufs ($n_0 = 350 \text{ U/min}$)

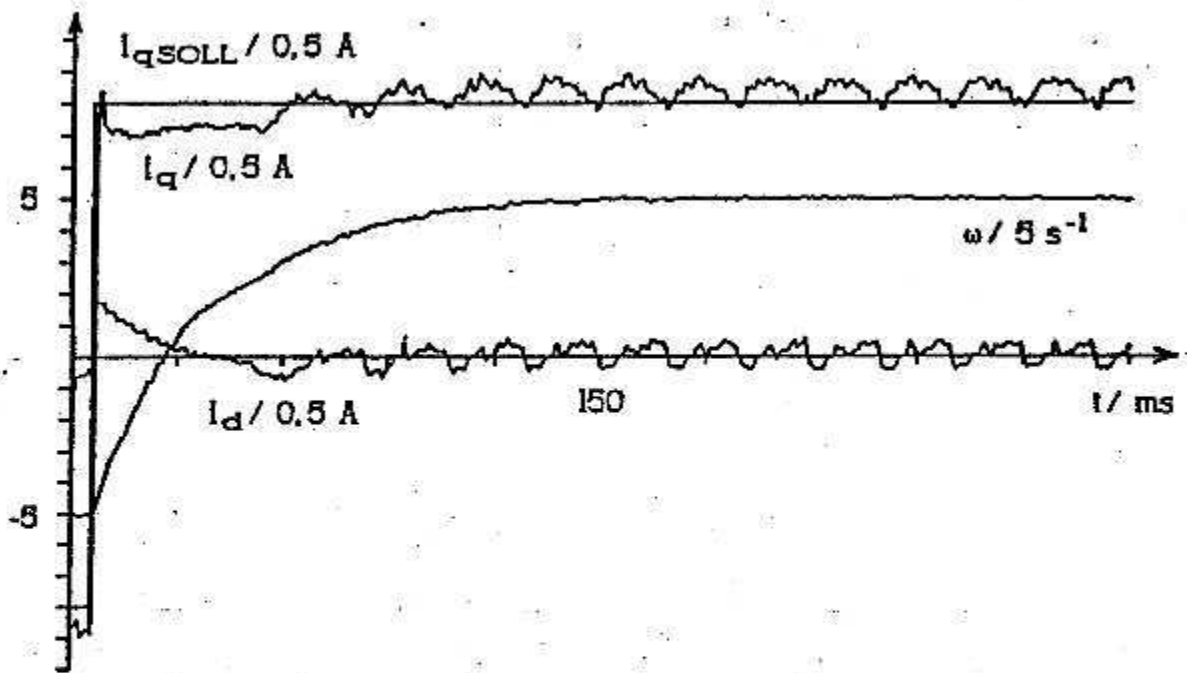


Bild 9.3.4 Stromregelung im rotorfesten System
Gemessener Stromverlauf bei einem Sollstromsprung

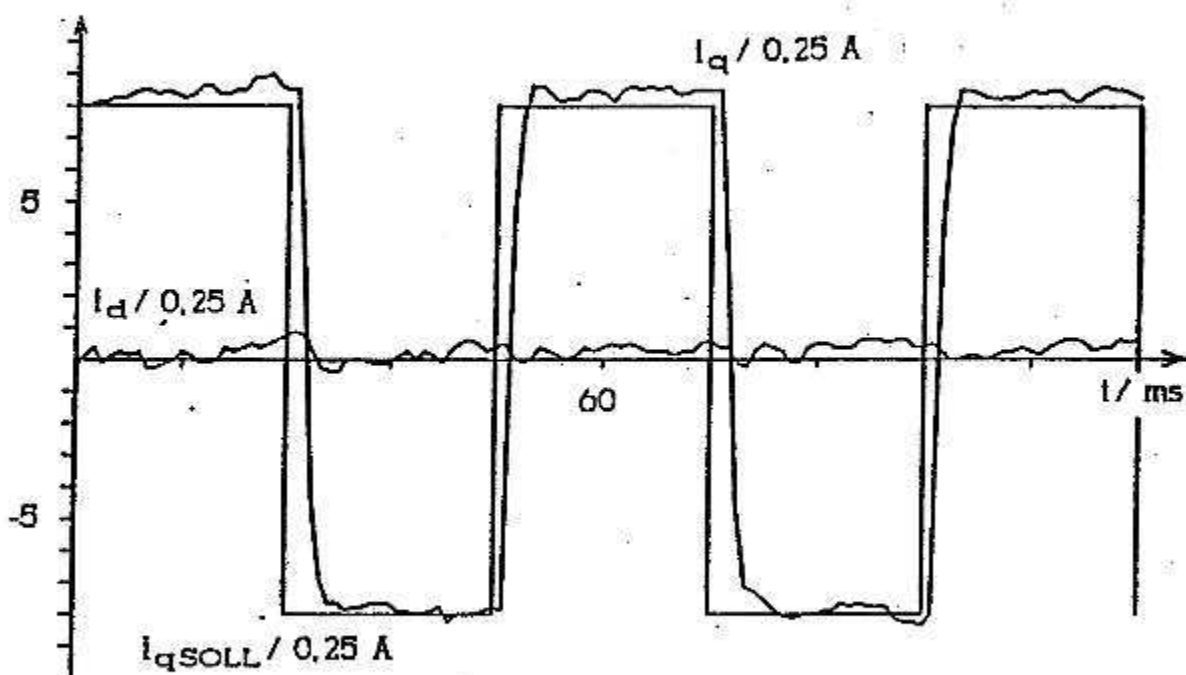


Bild 9.3.5 Stromregelung im rotorfesten System
Gemessener Stromverlauf bei einem rechteckförmigen Sollstromverlauf ($n_0 = 0 \text{ U/min}$)

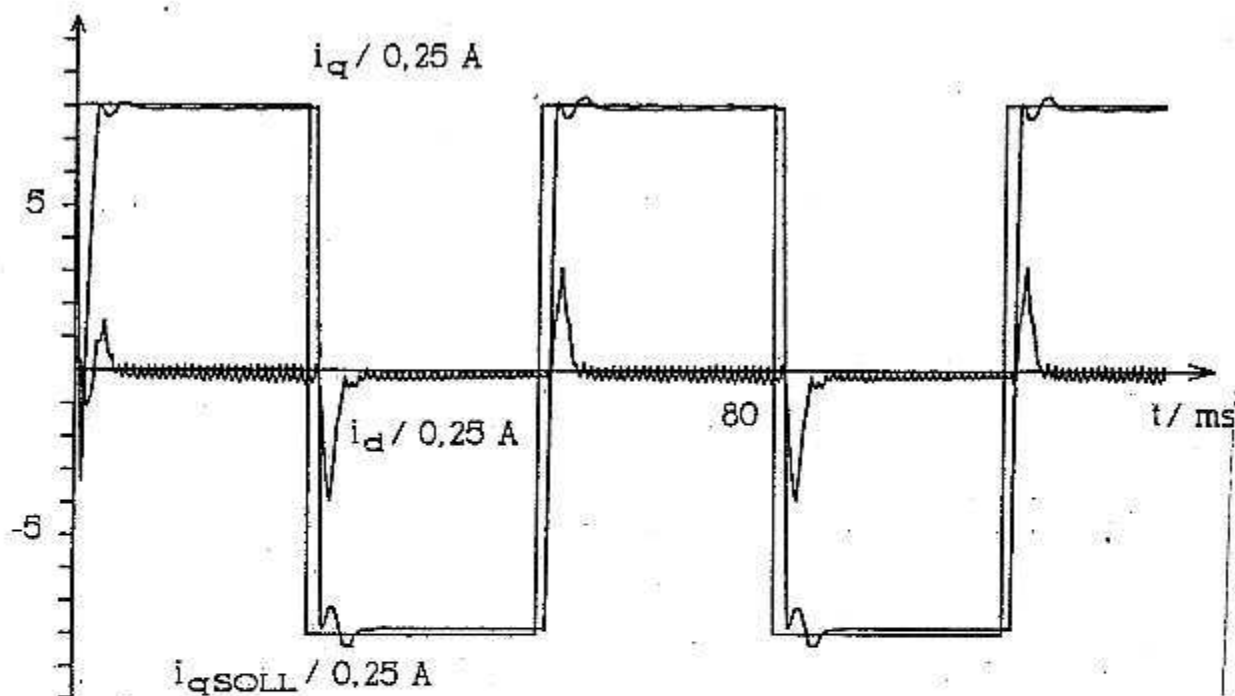


Bild 9.3.6 Stromregelung im rotorfesten System
Simulierter Stromverlauf bei einem rechteckförmigen Sollstromverlauf ($n_0 = 350 \text{ U/min}$)

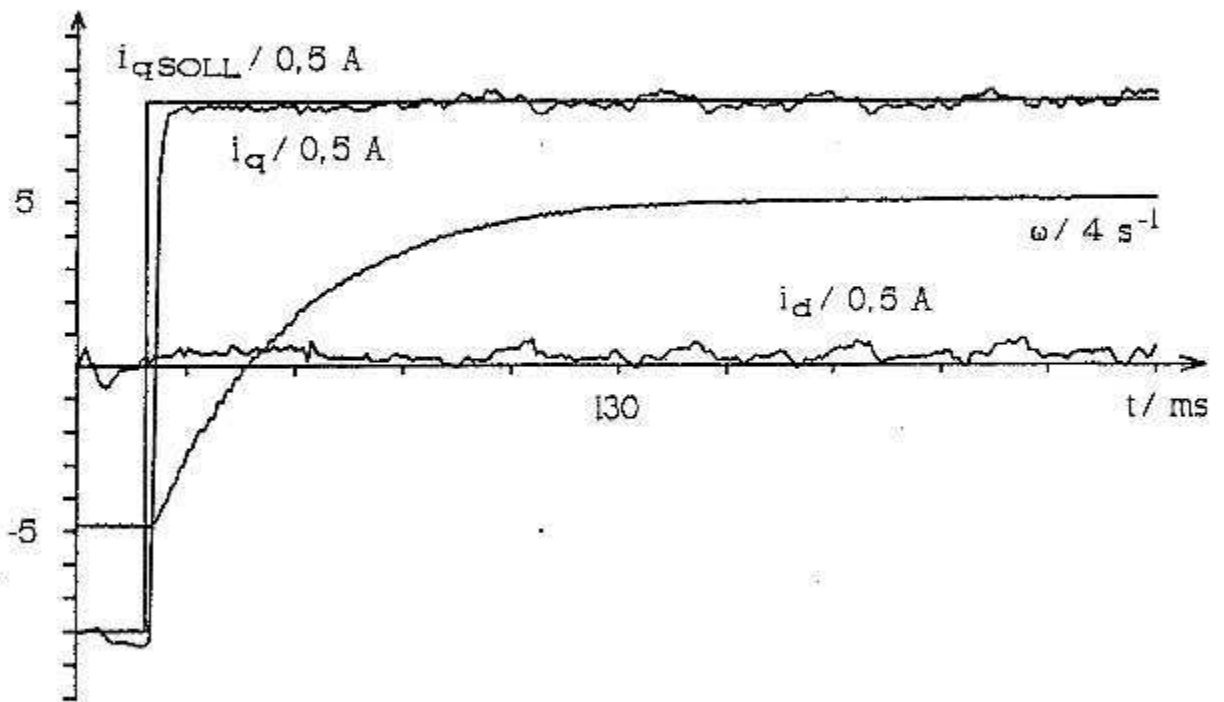


Bild 9.3.7 Stromregelung im statorfesten System
Gemessener Stromverlauf bei einem Sollstromsprung

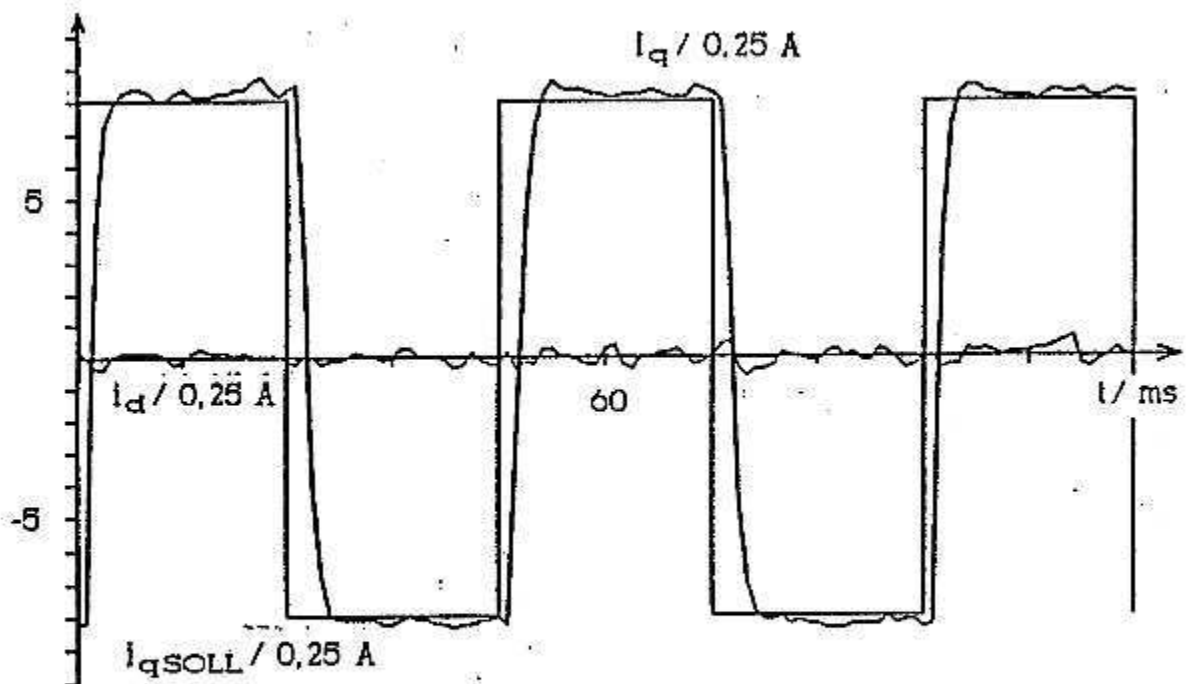


Bild 9.3.8 Stromregelung im statorfesten System
Gemessener Stromverlauf bei einem rechteck-
förmigen Sollstromverlauf ($n_0 = 0 \text{ U/min}$)

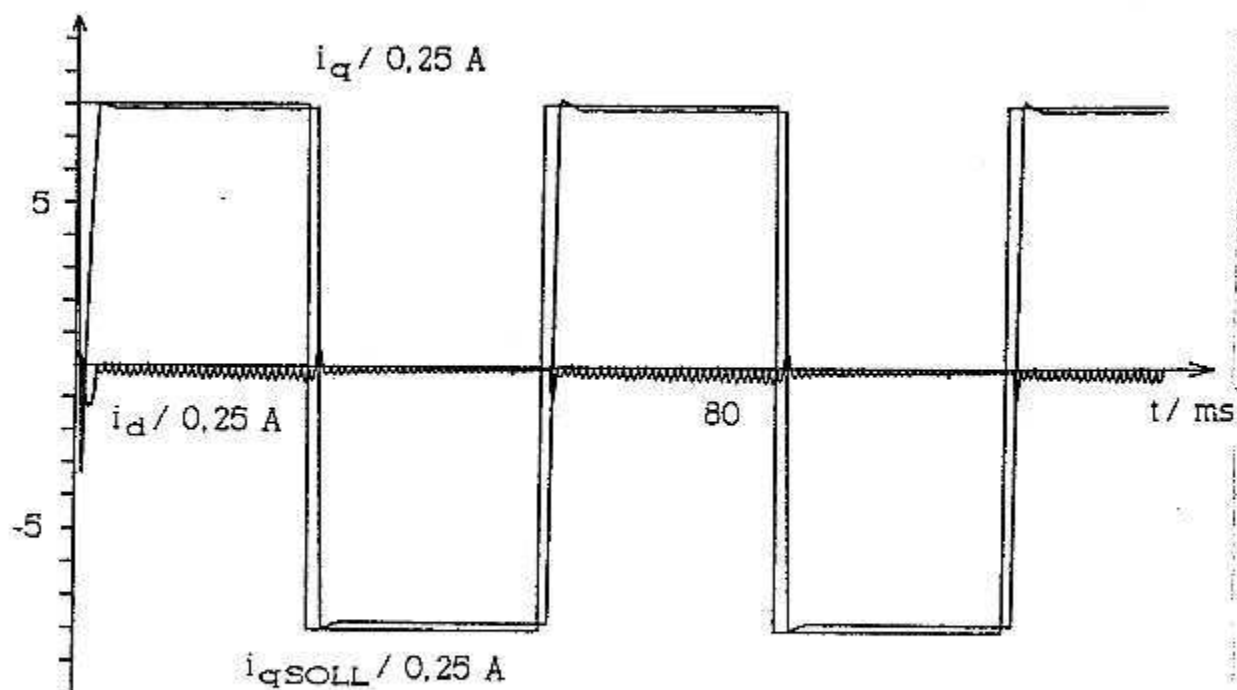


Bild 9.3.9 Stromregelung im statorfesten System
 Simulierter Stromverlauf bei einem rechteckförmigen Sollstromverlauf ($n_0 = 350 \text{ U/min}$)

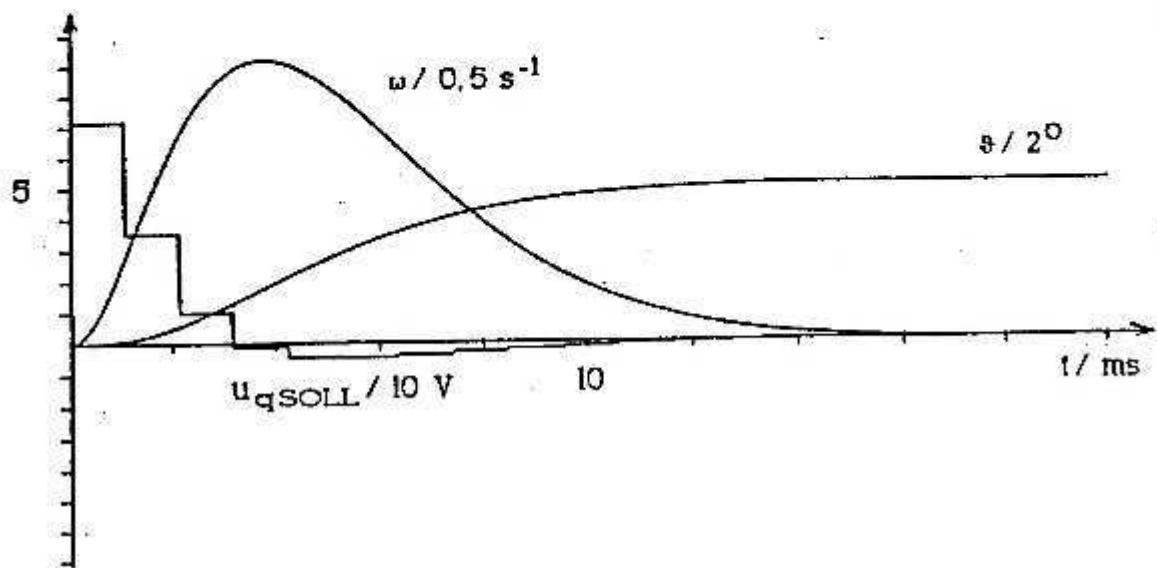


Bild 9.4.1 Simulierte Systemantwort auf einen
Lageführungssprung von 10°

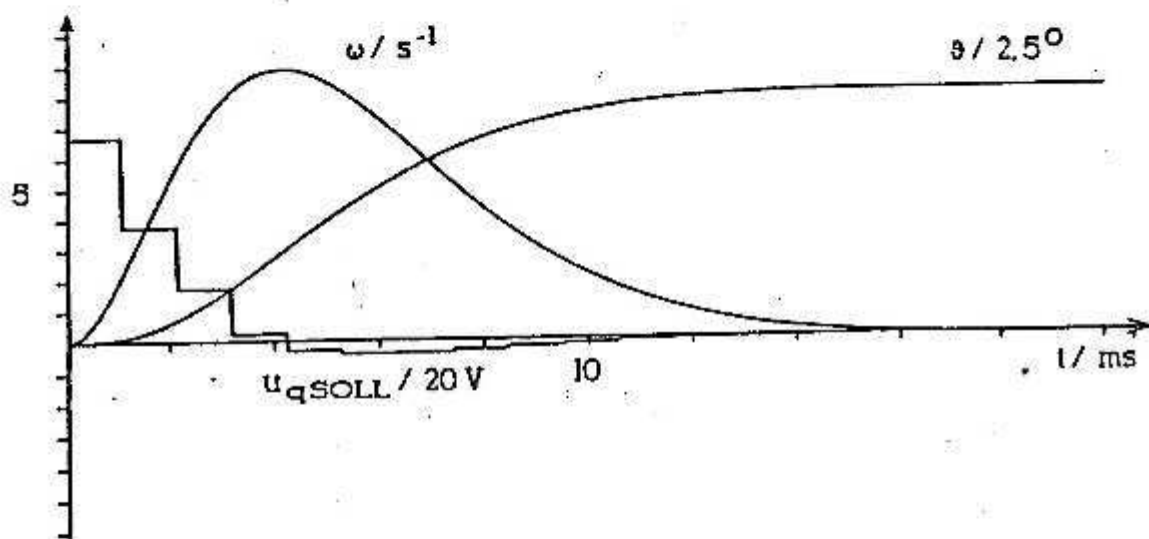


Bild 9.4.2 Simulierte Systemantwort auf einen
Lageführungssprung von 20°

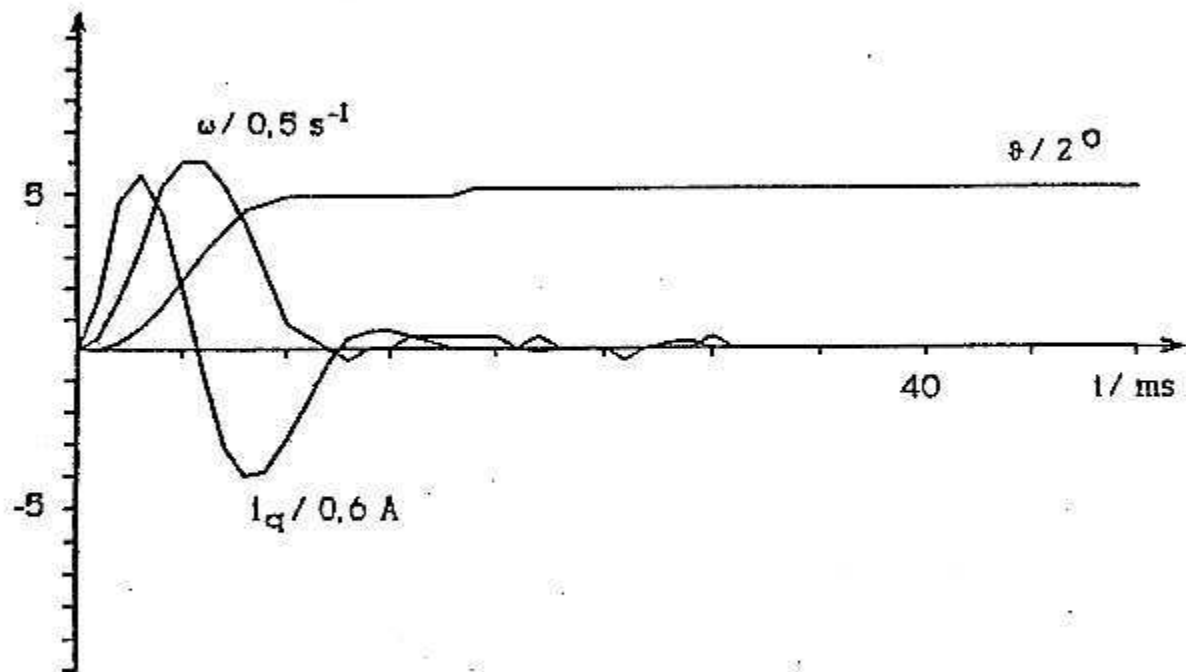


Bild 9.4.3 Gemessene Systemantwort auf einen
Lageführungssprung von 10°

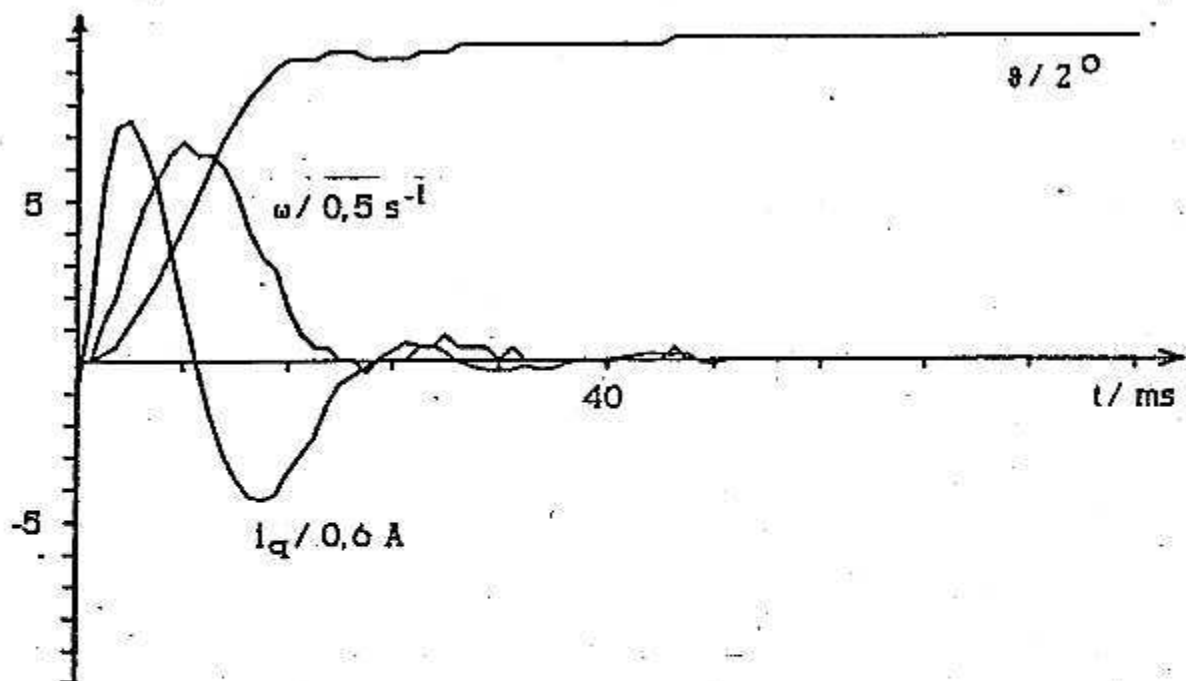


Bild 9.4.4 Gemessene Systemantwort auf einen
Lageführungssprung von 20°

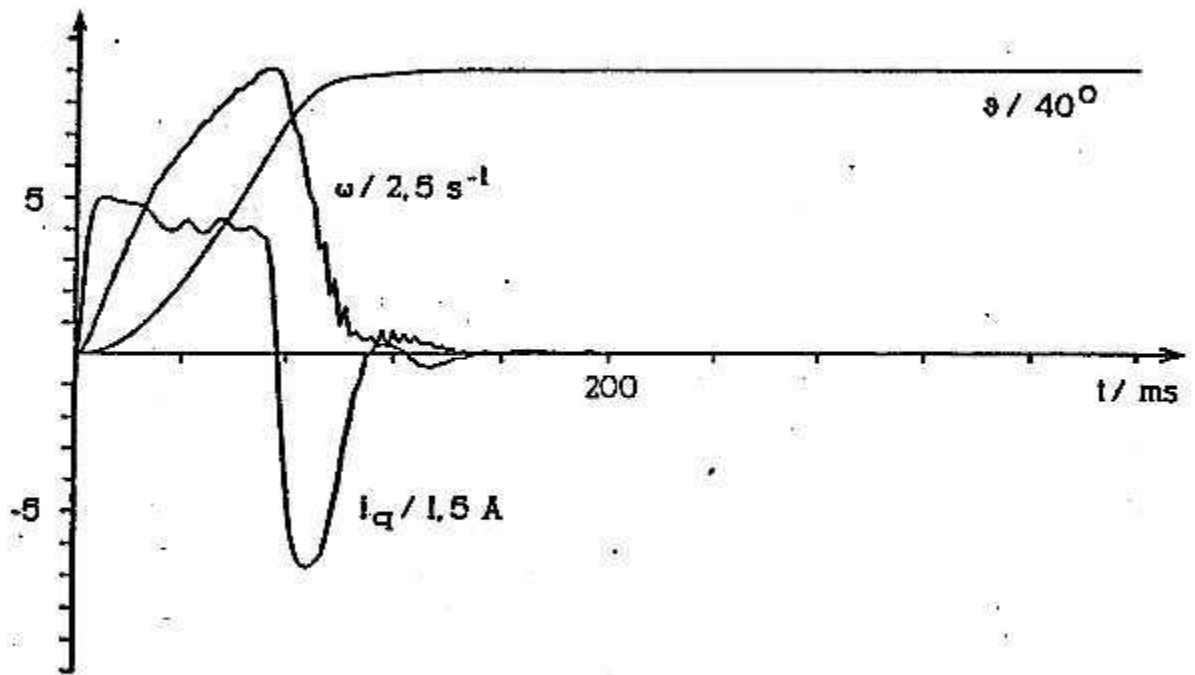


Bild 9.4.5 Gemessene Systemantwort auf einen Lageführungssprung von 360°

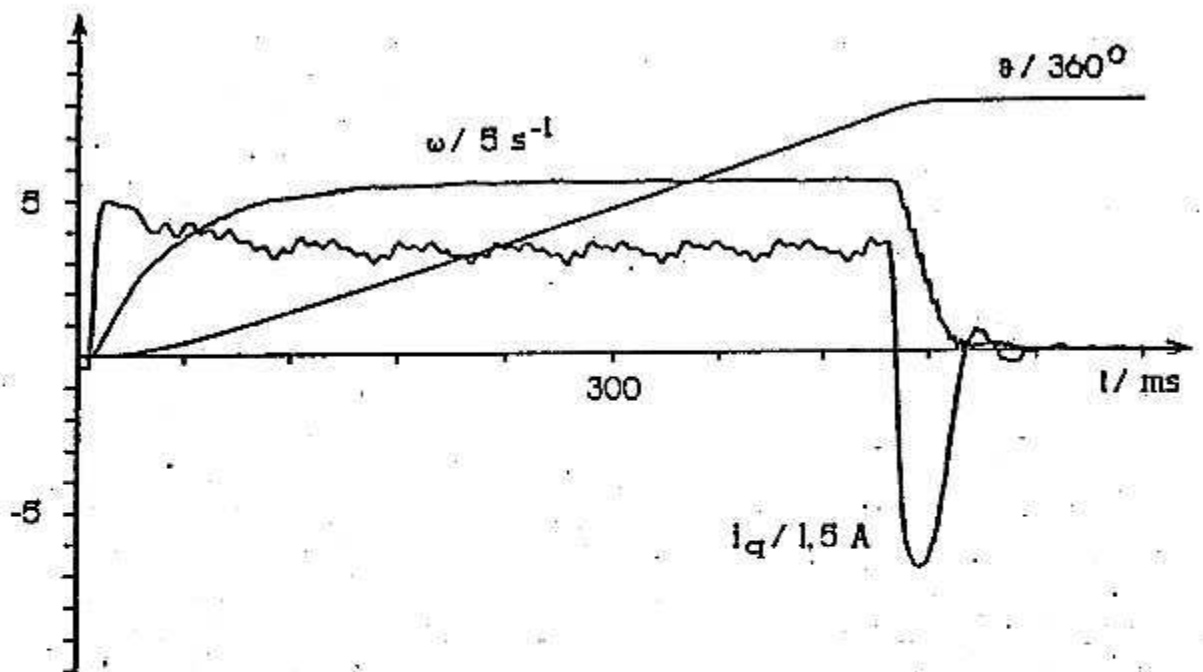


Bild 9.4.6 Gemessene Systemantwort auf einen Lageführungssprung von 8 Umdrehungen

ANHANG A: PROGRAMMDOKUMENTATION

A.1 DOKUMENTATION DES ASSEMBLERPROGRAMMS

Der Quellcode des Asemblerprogramms DSM18 hat einen Umfang von ca. 150 kB . Das Programm benutzt einige Assemblervariablen zur Steuerung der bedingten Assemblierung. Es stellt deshalb mehr einen Programmbaukasten als ein einfaches Programm dar. Die Notwendigkeit zur Verwendung der bedingten Assemblierung erwuchs aus der Beschränkung des für den Maschinencode zur Verfügung stehenden Speicherplatzes. Darüber hinaus wurde es als sinnvoll erachtet, stets nur mit einem Programm zu arbeiten. Dies erspart die bei der Arbeit mit mehreren Programmen notwendige Mehrarbeit. Die realisierten Maschinenprogramme haben einen Umfang von ca. 12 kB .

Die wichtigsten verwendeten Assemblervariablen haben die folgende Bedeutung:

TEST : Ist diese Variable gesetzt, so erfolgt eine erweiterte Fehlerprüfung während des Programmablaufs.

WA900 : Bei gesetzter Variablen erfolgt eine Umrechnung der vom Lagezähler gegebenen Winkelwerte auf 900 Incremente pro el. Umdrehung. Andernfalls erfolgt eine Umrechnung auf 1024 Incremente pro el. Umdrehung.

MAKRO : Bei gesetzter Variablen werden die Begrenzerrountinen und Transformationsrountinen als MACRO's realisiert. Andernfalls werden sie als Unterprogramme realisiert.

Da auch das Listing eines abgemagerten Programms, das von einem Programmbaukasten in Richtung eines einfachen Programms zurückentwickelt wurde, eine Länge von ca. 90 Seiten hat, wurde darauf verzichtet, das Listing in diese Arbeit aufzunehmen.

Statt dessen wird dieser Arbeit eine Diskette mit dem Quellcode beigelegt.

A.2 DOKUMENTATION DES SIMULATIONSPROGRAMMS

Der Quellcode des Simulationsprogramms SI16 hat einschließlich der verwendeten INCLUDE-Dateien einen Umfang von ca. 75 kB . Das erzeugte Maschinenprogramm hat einen Umfang von ca. 80 kB . Das Listing des Quellcodes hat eine Länge von ca. 55 Seiten. Anstelle des Listings ist dieser Arbeit eine Diskette mit dem Pascal-Quellcode beigelegt.

LITERATURVERZEICHNIS

- /1/ Buehler : Einführung in die Theorie geregelter Drehstromantriebe , Band 1
Birkhäuser Verlag, Basel 1977

- /2/ Grotstollen, Horst : Zusammenhänge und Kennwerte bei der Bemessung beschleunigungsoptimaler Antriebe
ETZ-A Bd. 95, 1974

- /3/ Hartmann ,Irmfried : Digitale Regelkreise
TUB-Dokumentation, TU-Berlin 1982

- /4/ Hetzel, Dieter : Permanentmagnetisch erregte Synchronmaschine - Maschinenmodelle, stationäres Verhalten und Steuerverfahren
Studienarbeit, TU Berlin 1989

- /5/ Marivor AC-Servomotor , Datenblatt
Fa. Marivor

- /6/ Leonhard, Werner : Control of Electrical Drives
Springer-Verlag, Berlin 1985

- /7/ Naunin, Dietrich : Skript zur Vorlesung "Dynamisches Verhalten von Stromrichter-Maschinen-Systemen"
TU Berlin, o.J.

- /8/ Nehl, Thomas u. a. : Dynamic simulation of radially oriented permanent magnet-type electronically operated synchronous machines with parameters obtained from finite element field solutions
IEEE Transactions, Vol. IA-18 , März 1982

- /9/ Orlik, Bernd : Zum Betriebsverhalten mikrorechnerge-
regelter und pulswechselrichter gespeister Synchron-
maschinen mit Permanentmagneterregung
Dissertation, TU Braunschweig 1986
- /10/ Schönfeld, Rolf : Digitale Regelung elektrischer
Antriebe
VEB Verlag Technik, Berlin 1987
- /11/ Cevik, Kemal : Ein Beitrag zur vollständig digitalen
Lagerregelung von Servoantrieben ohne Tachogenerator
am Beispiel eines Industrieroboters mit verteilter
Intelligenz
Dissertation, TU Berlin 1987